Universidade de Aveiro Departamento de Matemática
2009

Ana Luísa Romão de São Marcos

Avaliação de metodologias de pré-processamento de dados de *microarrays*

Universidade de Aveiro Departamento de Matemática
2009

Ana Luísa Romão de São Marcos

Avaliação de metodologias de pré-processamento de dados de *microarrays*

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Matemática e Aplicações, área de especialização Matemática Empresarial e Tecnológica, realizada sob a orientação científica da Profª. Doutora Adelaide de Fátima Baptista Valente Freitas, Professora Auxiliar do Departamento de Matemática da Universidade de Aveiro, e co-orientação científica da Profª. Doutora Gladys Castillo Jordán, Professora Auxiliar do Departamento de Matemática da Universidade de Aveiro.

à minha família, por todo o amor e em especial pelas longas conversas

o júri

presidente

Doutor Manuel González Scotto
Professor Auxiliar da Universidade de Aveiro

Doutora Adelaide de Fátima Baptista Valente Freitas
Professora Auxiliar da Universidade de Aveiro

Doutora Gladys Castillo Jordán
Professora Auxiliar da Universidade de Aveiro

Doutora Marília Cristina de Sousa Antunes
Professora Auxiliar da Universidade de Lisboa

agradecimentos

Em primeiro lugar, gostaria de agradecer às minhas orientadoras, a Profª. Doutora Adelaide de Fátima Baptista Valente Freitas e à Profª. Doutora Gladys Castillo Jordán, pela paciência que sempre tiveram comigo em ajustar-se ao meu ritmo de trabalho, fruto da distância física que nos separou durante meio ano, e que tornou difícil a conciliação entre as responsabilidades profissionais numa instituição europeia num país estrangeiro e a escrita desta dissertação. Este agradecimento não esquece também o acompanhamento, a competência, o rigor e a disponibilidade de ambas e os seus contributos para o enriquecimento deste trabalho.

Gostaria, também, de deixar uma palavra de agradecimento ao Miguel Monsanto do Biocant, que sempre se disponibilizou para deixar a processar nos computadores desse instituto as bases de dados, de grande volume, usadas neste estudo. À Laura Carreto, que se mostrou sempre muito prestável para responder a todas as minhas questões da área da Biologia e pela revisão desta dissertação no que respeita aos conceitos biológicos envolvidos, bem como pela oportunidade de poder estar presente numa experiência de *microarrays*.

Para finalizar, gostaria de agradecer à minha família. Ao Pedro, o meu companheiro no amor, pela troca de ideias e sugestões, revisão e apoio incondicional durante a realização deste trabalho. Aos meus pais, Jorge e Luísa, pelo apoio, confiança, motivação e inspiração dados. À minha irmã, Ana Jorge, pela boa disposição e carinho sempre partilhados. A eles dedico este trabalho.

palavras-chave

Dados de *microarrays* de expressão genética, classificação supervisionada, classificação de cancro, correcção de *background*, normalização, selecção de genes.

resumo

Esta dissertação surge no contexto da avaliação de metodologias de pré-processamento de dados de *microarrays* através do desempenho preditivo de modelos de classificação supervisionada.

As experiências de *microarrays* envolvem muitos passos, desde a extracção do tecido em estudo, passando pela marcação do mesmo com compostos fluorescentes, *scanning*, processamento de imagem, entre outras. Cada uma dessas etapas pode introduzir variabilidade nos dados recolhidos e assim afectar a qualidade dos mesmos.

Os métodos de pré-processamento de correcção de *background* (CB) e de normalização (NM) surgem da necessidade de remover as variações não desejadas mantendo as variações biológicas intrínsecas aos dados.

Para o presente trabalho foi realizado um estudo experimental onde foram aplicados aos dados vários métodos de CB e de NM, individualmente ou em conjunto, com a finalidade de avaliar o contributo destas metodologias no melhoramento da qualidade dos dados.

Apresenta-se aqui uma avaliação de 36 métodos pré-processamento (resultantes de combinações de métodos de CB e de NM) com base no desempenho preditivo de dois modelos de classificação, k-Vizinhos mais Próximos (k-NN) e Maquinas de Suporte Vectorial (MSV). Estes modelos são induzidos de três bases de dados públicas de microarrays de ADN-complementar, onde um par de métodos de pré-processamento, constituído por um de CB e outro de NM, é aplicado. A capacidade preditiva dos dois modelos de classificação é medida em termos da taxa de erro obtida pelo método de validação cruzada *leave-one-out*.

Em virtude da grande dimensão dos dados de microarrays, resultante de um elevado número de atributos (genes) envolvidos, o presente trabalho também inclui um estudo sobre o efeito da aplicação dos métodos de CB e de NM no desempenho preditivo de classificadores de MSV quando estes são induzidos de dados constituídos apenas por subconjuntos de genes altamente discriminativos.

keywords

Microarray gene expression data, supervised classification, cancer classification, background correction, normalization, gene selection.

abstract

This dissertation addresses the problem of evaluating preprocessing methodologies in terms of the predictive performance of supervised classification models induced from microarray data.

Microarray experiments involve many steps, from the extraction of the tissue in study, through its labeling with fluorescent dyes, scanning and image processing, among others. Each of these stages can introduce variability in the data collected and thus affect their quality.

Preprocessing methods such as background correction (BC) techniques and normalization (NM) strategies have arisen from the need to remove the unnecessary variation while the intrinsic biological variations of the data are retained.

In this work an experimental study has been carried out where various BC and NM methods have been employed on the data, individually or in combination, with the goal of assessing the contribution of these approaches to the improvement of the quality of the data.

Herein is presented an evaluation of 36 preprocessing methods (resulting from combinations of BC and NM methods) in terms of the predictive performance of two classification models, k-Nearest Neighbours (k-NN) and Support Vector Machines (SVM). These models are induced from three publicly available cDNA microarray data sets, where a pair of preprocessing strategies, composed of a BC technique and a NM method, is employed. The predictive performance of both classifiers is measured on grounds of the error rate obtained by the leave-one-out cross validation method.

Due to the high dimensionality of microarray data, resulting from a large number of attributes (genes) involved, this dissertation also includes a study about the effect of the application of BC and NM methods on the predictive performance of SVM classifiers when these are induced from data consisting of only subsets of highly discriminative genes.

Dor de alma

Meu pratinho de arroz doce
polvilhado de canela!
Era bom mas acabou-se
desde que a vida me trouxe
outros cuidados com ela.

Eu, infante, não sabia
as mágoas que a vida tem.
Ingenuamente sorria,
me aninhava e adormecia
no colo da minha mãe.

Soube depois que há no mundo
umas tantas criaturas
que vivem num charco imundo
arrancando o arroz do fundo
de pestilentas planuras.

Um sol de arestas pastosas
cobre-os de cinza azebre
à flor das águas lodosas,
eclodindo em capciosas
intermitências de febre.

Já não tenho o teu engodo,
ó mãe, nem desejo tê-lo.
Prefiro o charco e o lodo.
Quero o sofrimento todo.
Quero senti-lo, e vencê-lo.

António Gedeão

Conteúdo

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas

ADN	Ácido Desoxirribonucleico
ARN	Ácido Ribonucleico
CB	Correcção de *Background*
DCBD	Descoberta de Conhecimento em Bases de Dados
DLBCL	*Diffuse large B-cell lymphoma*
k-NN	k-Nearest Neighbour
loess	*Local weighted regression*
LOO-CV	*Leave-one-out cross-validation*
MSV	Máquinas de Suporte Vectorial
NM	Normalização
PT	*print-tip*
SSA	Selecção de subconjuntos de atributos
TE	Taxa de erro
TR	Taxa de redução
TRM	Taxa de redução média

Capítulo 1

Introdução

1.1 Contextualização

Em 1953, Francis Crick e James Watson descobriram a estrutura do ADN[1] e a forma como este codifica as proteínas. Estas descobertas, em conjunto com os primeiros passos na sequenciação de genomas iniciados na década de 90, têm contribuído significativamente para o progresso da Genética, em particular, da Genética Funcional.

O desenvolvimento da Biotecnologia, nomeadamente a tecnologia de *microarrays*, onde milhares de genes[2] podem ser monitorizados simultaneamente, tem possibilitado o avanço na investigação de propriedades de diversos genes. Esta tecnologia foi desenvolvida no início da década de 90 pelo grupo de investigação da Universidade de Stanford nos EUA [57]. Trata-se de uma ferramenta amplamente utilizada na área da Biologia e da Medicina e permite investigar o nível de resposta de milhares de genes face a variações de condições experimentais específicas [30].

Os dados extraídos de experiências de *microarrays*, em geral, precisam de refina-

[1] O ácido desoxirribonucleico (abreviadamente, ADN; em inglês DNA) é um polímero longo duplamente ligado onde cada cadeia é constituída por sequências de bases nitrogenadas ou nucleótidos. Para além destes nucleótidos existem outros três componentes: pentose, grupo fosfato e pontes de hidrogénio. As bases podem ter quatro variantes: A-Adenina,T-Timina,C-Citosina,G-Guanina.

[2] Um gene é uma sequência de ADN que codifica uma proteína específica (*i.e.* uma sequência de aminoácidos)

mento em virtude de decorrerem de procedimentos experimentais reais sem qualquer recorrência a técnicas de simulação controladas. Habitualmente são aplicadas, a este tipo de dados, métodos de pré-processamento específicos antes da análise dos mesmos.

A presente dissertação de mestrado insere-se no âmbito do projecto de investigação da Universidade de Aveiro, "Novas metodologias estatísticas para análise de dados de *microarrays* de ADN"(PTDC/MAT/72974/2006), financiado pela Fundação para a Ciência e Tecnologia (FCT). Uma das tarefas desse projecto relacionou-se com o estudo do efeito de metodologias de pré-processamento em dados de *microarrays* de ADN-complementar. Pretende-se com o presente trabalho dar um contributo à concretização dessa tarefa.

Neste capítulo introdutório desenvolve-se a noção de *microarrays* e os procedimentos experimentais necessários na utilização dessa tecnologia. São ainda referidos os métodos de correcção de *background* e os métodos de normalização, dois tipos de metodologias de pré-processamento mais aplicadas em dados de *microarrays*. Posteriormente, faz-se uma breve introdução ao procedimento aqui usado na avaliação de métodos de pré-processamento, o qual tem por base modelos de classificação supervisionada. De seguida, expõem-se os objectivos gerais deste trabalho, e por fim, resumem-se os diferentes capítulos que fazem parte integrante desta dissertação.

1.2 Conceitos básicos

1.2.1 *Microarrays*

A tecnologia de *microarrays* é uma ferramenta sem precedentes para a recolha de enormes quantidades de dados de expressão genética numa única experiência. Em termos físicos, um *microarray* consiste numa superfície rectangular sólida revestida

por milhares de pontos microscópicos de ADN oligonucleótido[3] contendo cada um sequências de ADN específicas. Esta superfície pode também ser vista como uma matriz ordenada de pontos agrupados em grupos-PT (traduzido do termo em inglês *print-tip-groups*) e dispostos de acordo com uma configuração definida por quatro parâmetros: ngl - número de grupos-PT por linha, ngc - número de grupos-PT por coluna, npl - número de pontos por cada linha de um grupo-PT, npc - número de pontos por cada coluna de um grupo-PT. Por exemplo, a Figura 1.1 ilustra um *microarray* com uma configuração ngr = 2, ngc = 2, nsr = 12, nsc = 12, ou seja, composto por quatro grupos-PT dispostos em 2 linhas e 2 colunas onde cada grupo-PT é representado por uma grelha de 12×12 pontos.

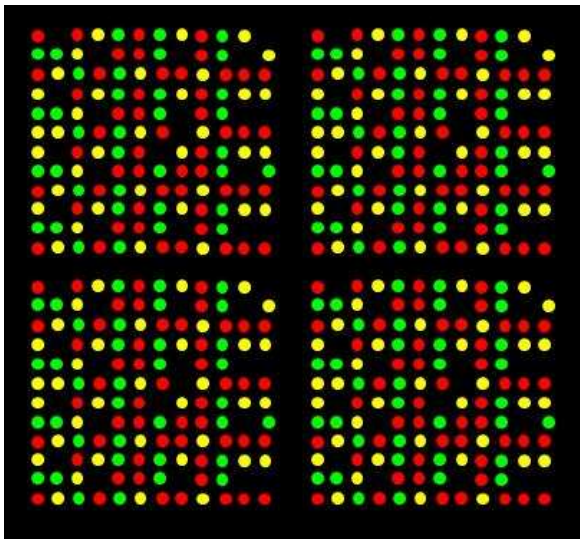


Figura 1.1: Representação de um registo digital das fluorescências emitidas em cada ponto de um *microarray* de ADN-complementar após ser processado por um *scanner*.

Nas experiências de *microarrays* é medido o nível de expressão de fragmentos de ADN ou genes de amostras de tecidos, sob certas condições experimentais específicas. A expressão genética é o processo de transcrever a informação codificada nos genes em

[3]Um oligonucleótido é um fragmento curto de uma cadeia simples de ácido nucleico, tipicamente com 20 ou menos bases. O ácido nucleico pode ser o ADN ou o ácido ribonucleico (abreviadamente, ARN; em inglês, RNA). O ARN distingue-se do ADN por ser uma molécula de cadeias simples, onde o nucleótido Timina é substituído pelo nucleótido Uracilo e a desoxirribose pela ribose. Os oligonucleótidos são frequentemente usados como sondas para detectar ADN-complementar ou ARN porque ligam-se prontamente aos seus complementares.

ARN e sua subsequente tradução em proteínas. O número de cópias de ARN de um gene indica aproximadamente o *nível de expressão* desse gene, ou seja, a quantidade de proteína correspondente existente na célula. Um dos principais objectivos neste tipo de experiência é a identificação de genes importantes de entre os muitos para os quais as medidas de expressão foram obtidas. A noção de importância corresponde à associação com uma resposta de interesse. Quando se contrastam níveis de expressão de um gene, ou seja, quando o nível de expressão de um gene muda significativamente entre duas condições experimentais específicas, esses genes importantes são designados de *diferencialmente expressos* [71].

Os procedimentos na execução de uma experiência de *microarrays* dependem da tecnologia utilizada, podendo esta variar na forma como os fragmentos de ADN são impressos ou ainda se são *microarrays* de um ou de dois canais. O Affymetrix GeneChip é um caso particular da tecnologia de *microarrays* oligonucleótidos (*oligonucleotide microarrays*) de um canal [42]. A presente dissertação analisa dados de *microarrays* de ADN-complementar que são do tipo *spotted microarray*, em particular usaram-se dados de dois canais [4]. []

Na Figura 1.2 encontra-se esquematizado o procedimento de recolha de dados de uma experiência de *microarrays* de dois canais. Este inicia-se com a extracção de ARN mensageiro[5] de dois tipos de tecidos (Tipo A e Tipo B: *e.g.* normal e canceroso), que se designam, em inglês, por *targets*. Posteriormente, o ARN mensageiro dos dois tecidos passa por um processo de transcrição reversa e é convertido em ADN-complementar. De seguida, cada ADN-complementar é marcado com um composto fluorescente (ou fluoróforo), um tecido com fluoróforo Cy5 e o outro com fluoróforo Cy3. As duas amostras de tecidos são misturadas e colocadas sobre o *microarray* onde estão impressas em diferentes pontos sequências de nucleótidos, tecnicamente designadas por sondas. Neste passo dá-se a hibridação competitiva entre as sondas e os *targets* por um período de incubação apropriado. Terminado o tempo de incubação, o *microarray* é lavado

[4]No decorrer deste trabalho serão utilizadas várias expressões para também designar *microarray*, estas são, lâmina de vidro, lâmina rectangular e, apenas, lâmina.

[5]O ARN pode aparecer em três diferentes formas onde o ARN mensageiro é a sua versão mais importante, tendo este a função de transportar a mensagem genética desde o ADN até ao ponto onde é traduzida no citoplasma.

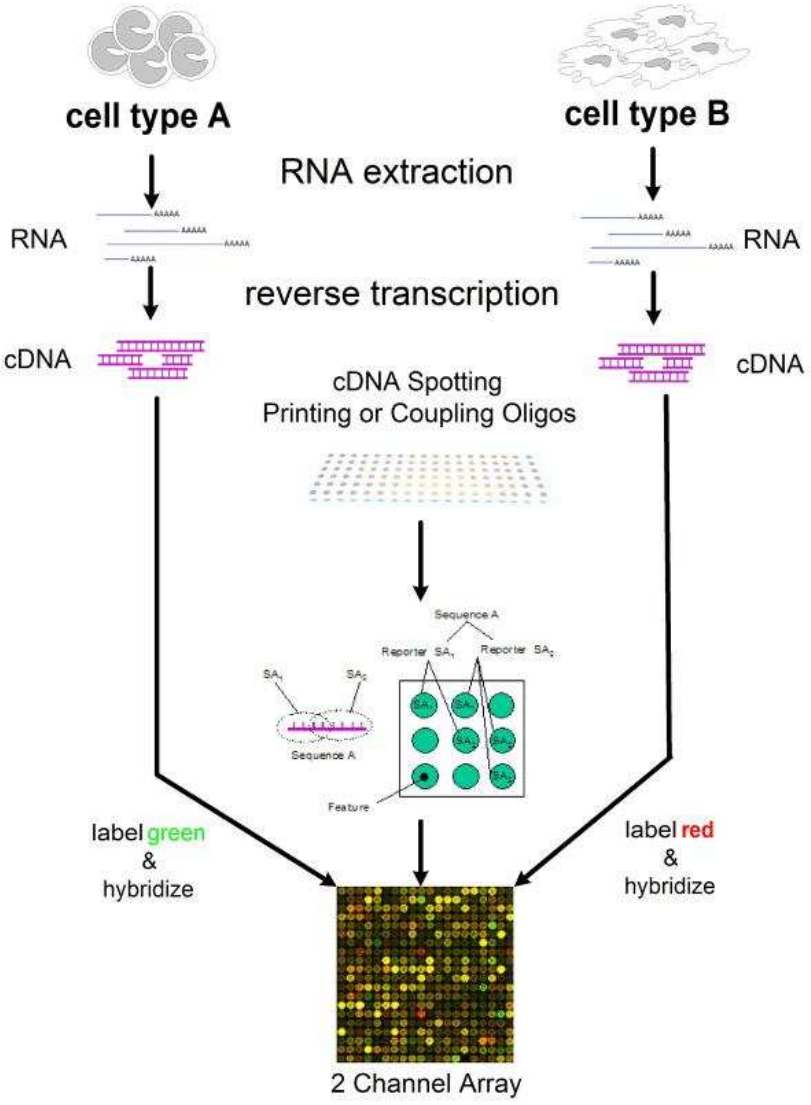


Figura 1.2: Fluxograma da parte laboratorial de uma experiência de *microarrays* de ADN-complementar de dois canais (figura extraída de [79]).

para retirar o excesso de amostra que não hibridou com as sondas.

A lâmina do *microarray* é processada por um *scanner* que, ao emitir um laser com determinado comprimento de onda, excita os fluoróforos Cy3 e Cy5. A quantidade de fluorescência emitida aquando da emissão do laser corresponde à quantidade de ADN ligado em cada ponto do *microarray*. A imagem resultante do *scanner* é obtida em tons de cinzento, na realidade existem duas imagens, uma para o Cy3 e outra para o Cy5 que são sobrepostas pelo *software* de processamento de imagem. É o *software* que dá a gradação de cores habitual, em verde e vermelho, assim escolhida por ser intuitivo a relação entre as duas cores e razão das duas fluorescências. Se, na imagem obtida depois de aplicado o *software* de processamento de imagem, o ADN do tecido marcado com o fluoróforo Cy5 está em abundância num ponto do *microarray*, a imagem reproduzida pelo *scanner* mostrará o ponto representado com cor vermelha, enquanto que se o ADN do tecido marcado com composto fluorescente Cy3 está em abundância a imagem regista o ponto representado com cor verde. No caso da quantidade de ambos os fluoróforos ser igual a imagem exibe o ponto representado com cor amarela e, por último, se não há presença de ADN não haverá fluorescência e o ponto aparecerá representado com preto na imagem, ver Figura 1.1.

De seguida, são lidas as intensidades de cada sonda para os canais representados com verde e vermelho, originando um valor numérico que define o nível de expressão de cada sonda ou gene. Uma explicação mais detalhada sobre o procedimento de uma experiência de *microarrays* de ADN-complementar, pode ser encontrada no portfolio[6] realizado pelo grupo de Bioinformática da Universidade de Aveiro no seu sítio na Internet[7] e em [24, 27, 52].

1.2.2 Pré-processamento

O grande volume de informação recolhida em apenas uma única experiência de *microarrays*, tem proporcionado a aplicação de um número considerável de metodologias estatísticas. As experiências de *microarrays* envolvem muitos passos, *e.g.* impressão do

[6]Este portfolio foi realizado no âmbito do projecto *Novas metodologias Estatísticas para Análise de dados de Microarrays de ADN*.

[7]`http://bioinformatics.ua.pt/resources/pub/pfma.pdf`

ADN-complementar, extracção de ARN mensageiro, marcação, hibridação, *scanning*, processamento de imagem, entre outras. Cada uma dessas etapas pode introduzir variabilidade nas intensidades medidas e assim afectar a qualidade dos dados recolhidos [30]. Assim, os dados brutos de *microarrays* são influenciados por variações de circunstâncias imprevisíveis que, na sua maioria, dependem de factores técnicos.

Os métodos de correcção de *background* (CB) e de normalização (NM) são dois tipos de métodos de pré-processamento destinados a refinar dados brutos em experiências de *microarrays*. Deste tipo de dados é expectável que o efeito de variações não desejadas sejam retiradas, mantendo as variações biológicas intrínsecas aos mesmos [76].

Em particular, os métodos de CB são aplicados com o objectivo de remover o ruído de fundo não específico da intensidade total medida pelo *scanner*. Em [50, 67] é referido que a remoção inapropriada de *background* local pode introduzir mais ruído nos dados. Assim, a implementação de técnicas de CB sobre dados brutos de microarrays é ainda um tema em estudo.

Por sua vez, os métodos de NM pretendem remover as variações que advêm de fontes aleatórias, como as diferenças de eficiência entre os dois fluoróforos Cy3 e Cy5 no processo de marcação.

Neste estudo foram aplicados aos dados métodos de CB e de NM, individualmente ou em conjunto, com a finalidade de avaliar o contributo destas metodologias no melhoramento da qualidade dos dados.

1.2.3 Classificação supervisionada de dados de expressão genética

Usualmente, a enorme quantidade de informação obtida das experiências de *microarrays* é organizada numa matriz $m \times n$ de níveis de expressão genética, onde m representa o número de amostras de tecidos e n o número de genes. Além da matriz de níveis de expressão genética é também conhecida a classe de cada amostra do tecido no que respeita a sua classificação. Por exemplo, para o problema de classificação de cancro, a classe pode ser *tecido canceroso* ou não.

Nos últimos anos, houve uma interessante abordagem da avaliação de métodos de pré-processamento, aplicados a dados de *microarrays*, no contexto de aprendizagem

supervisionada [76]. A aprendizagem supervisionada traduz a tarefa de induzir uma função capaz de predizer com alta fiabilidade as classes de futuros objectos, descritos por um conjunto de atributos, a partir de um conjunto de exemplos rotulados. No caso do problema em estudo, e tendo em conta as designações acima referidas, os exemplos denotam as amostras de tecido. Estas amostras são descritas por atributos que, por sua vez, representam os genes, e a classe predefinida designa a classe do tecido, *e.g.* canceroso ou não.

Os dados de *microarrays* têm características que os tornam muito peculiares na tarefa de classificação, nomeadamente a sua grande dimensão, geralmente muitos milhares de genes, e um número muito reduzido de amostras de tecidos, raramente ultrapassando a centena. Estas características têm levado a um grande esforço por parte dos investigadores em se encontrarem metodologias para reduzir essa elevada dimensionalidade, um problema conhecido como *selecção de genes*. Apesar da elevada dimensão intrínseca a este tipo de dados, foi provado em [28, 72] que um pequeno subconjunto de genes altamente discriminativos é suficiente para construir classificadores bastante precisos. No entanto, a precisão de previsões futuras pode ainda depender de outros factores, designadamente da implementação de métodos de pré-processamento apropriados.

Os estudos já elaborados relacionados com a avaliação de metodologias de pré-processamento para dados de *microarrays* têm sido mais direccionados para a avaliação dos métodos de NM [69, 76] do que CB [53, 56]. Os métodos de NM têm sido avaliados usando critérios, como por exemplo, a habilidade de detectar genes diferencialmente expressos usando a razão de falsas descobertas [53], o erro quadrático médio [48], entre outros. Até à data da escrita desta dissertação, só se tem conhecimento de um trabalho que avalia a efectividade de métodos de NM em termos da capacidade preditiva dos classificadores induzidos de dados de *microarrays* [76].

A Figura 1.3 sintetiza os diversos passos necessários até à avaliação e comparação de métodos de pré-processamento de dados de *microarrays*. Na figura estão registados os passos desde a execução da experiência em laboratório até à avaliação de modelos de classificação induzidos dos dados aos quais dois tipos de metodologia de pré-processamento foram aplicados.

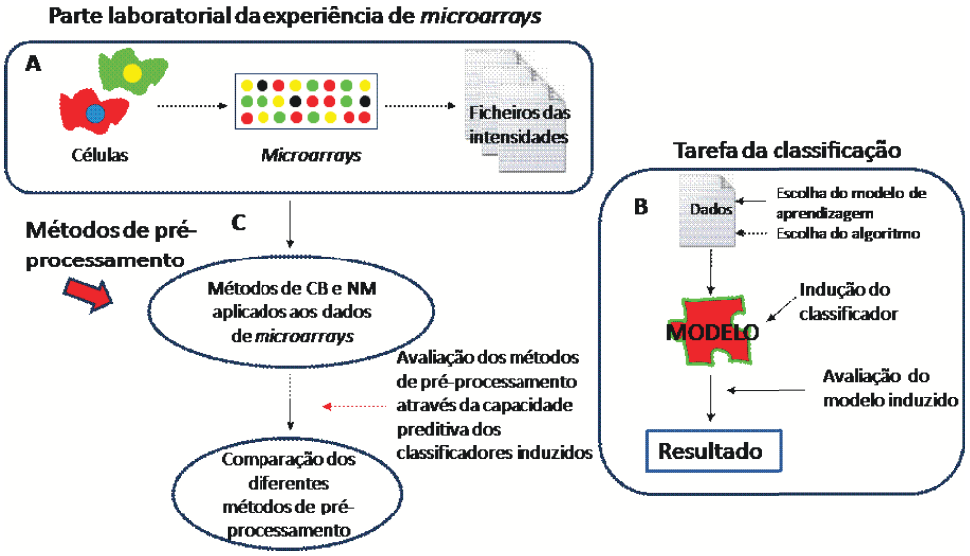


Figura 1.3: Fluxograma dos diferentes passos do processo de análise de dados de *microarrays*. **A.** Procedimento experimental de uma experiência de *microarrays* desde a recolha dos tecidos até à obtenção dos ficheiros de texto. **B.** Tarefa de classificação sumariada nos seus principais passos. **C.** Avaliação de métodos de pré-processamento.

1.3 Objectivos gerais

O objectivo desta dissertação é apresentar uma avaliação de métodos combinados de CB e de NM com base no desempenho preditivo de dois modelos de classificação, *k*-vizinhos mais próximos e máquinas de suporte vectorial. Estes modelos são induzidos de três bases de dados públicas de *microarrays* de ADN-complementar, onde um par de métodos de pré-processamento, composto por um de CB e outro de NM, é aplicado.

Tendo em conta a quantidade de dados a processar foi utilizado um novo *software* da área da aprendizagem automática, RapidMiner [44]. A escolha deste *software* prendeu-se com a grande capacidade de processamento e pela variedade de modelos e algoritmos disponíveis ao utilizador. Numa fase prévia do estudo foi ainda utilizado o pacote *Bioconductor* do *software R*, este último de acesso livre para computação estatística e gráfica [32]. Este *software* contém muitas livrarias implementadas com o intuito de facilitar o refinamento de dados brutos de experiências de *microarrays* e posterior análise.

1.4 Organização da dissertação

A presente dissertação é constituída, para além desta introdução, por três capítulos adicionais.
Começar-se-á no Capítulo 2 por descrever formalmente uma abordagem supervisionada do problema de classificação de cancro. Este problema de classificação particular, usando dados de expressão genética, tem grande importância na detecção de diferentes tipos de cancro e permite fornecer um diagnóstico específico a uma dada situação. Enunciar-se-ão os dois modelos de classificação supervisionada usados no presente estudo: *i)* k-vizinhos mais próximos e *ii)* máquinas de suporte vectorial. Existem vários métodos distintos para obter estimativas fiáveis sobre o desempenho de classificadores induzidos de dados. Neste capítulo estudar-se-á apenas a taxa de erro obtida pelo método de validação cruzada *leave-one-out* que é aplicado aos dois modelos de classificação estudados.

O Capítulo 3 começará com uma nota introdutória sobre o conceito de pré-processamento de dados. A correcção de *background* é primeiramente abordada e é feita referência aos tópicos de processamento de imagem e métodos de estimação de *background*. Posteriormente apresenta-se um estado da arte dos métodos de CB e definem-se cinco métodos CB estudados nesta dissertação. A secção sobre a normalização é iniciada com uma abordagem ao modelo de regressão *loess* e, seguidamente é apresentado um estado da arte dos métodos de NM. O capítulo será finalizado com uma explicação de cinco métodos de NM usados no estudo experimental aqui realizado.

O Capítulo 4 consta do principal objectivo da investigação que conduziu à elaboração da presente dissertação e que contribui para o projecto de investigação supramencionado. Concretamente, no Capítulo 4 apresentar-se-ão os resultados de dois estudos experimentais executados. O primeiro estudo mostrará os resultados das 36 combinações de métodos de CB e NM utilizando os dois modelos de classificação supervisionada destacados no Capítulo 2. O principal objectivo é comparar as diferentes estratégias combinadas. Para isso analisar-se-á o desempenho preditivo dos modelos de classificação supervisionada induzidos de dados de *microarrays*. Estes dados constam de um repositório de bases de dados de vários tipos de cancro referenciadas na literatura especializada. No segundo estudo tomam-se subconjuntos de genes altamente discrimi-

nativos, obtidos usando três critérios distintos, com o objectivo de avaliar até que ponto a remoção de ruído técnico através de métodos de CB e de NM influencia o processo de selecção de genes.

Finalmente, no Capítulo 5 apresentar-se-ão as conclusões, com um sumário das principais contribuições deste trabalho com consequências para a área da Bioinformática. Terminar-se-á ainda com as principais ideias para um futuro trabalho de investigação de temas que ficam em aberto nesta dissertação.

Capítulo 2

Classificação supervisionada de cancro

2.1 Introdução

O problema de classificação supervisionada de cancro é um tópico muito particular que se insere num conjunto de disciplinas mais abrangentes que vão desde a Inteligência Artificial, passando pela Descoberta de Conhecimento em Bases de Dados (DCBD)[1], Aprendizagem Automática[2] e ainda outros campos de investigação. O objectivo da Figura 2.1 é situar, em termos de disciplinas, o problema de classificação supervisionada de cancro. Para isso utilizam-se sobreposições de diversas áreas de modo que a intersecção final seja o problema em estudo.

A aprendizagem automática é uma área da disciplina da inteligência artificial que se centra no estudo e desenvolvimento de programas computacionais que automaticamente melhoram o seu desempenho através da experiência [45]. No contexto da aprendizagem automática é possível identificar diferentes classes de algoritmos de aprendizagem dependendo do resultado desejado, por exemplo, algoritmos de aprendizagem supervisionada, aprendizagem não supervisionada, entre outros [54].

[1] Tradução do termo inglês *Knowledge Discovery in Databases*, em abreviatura KDD.
[2] Tradução do termo inglês *Machine Learning*.

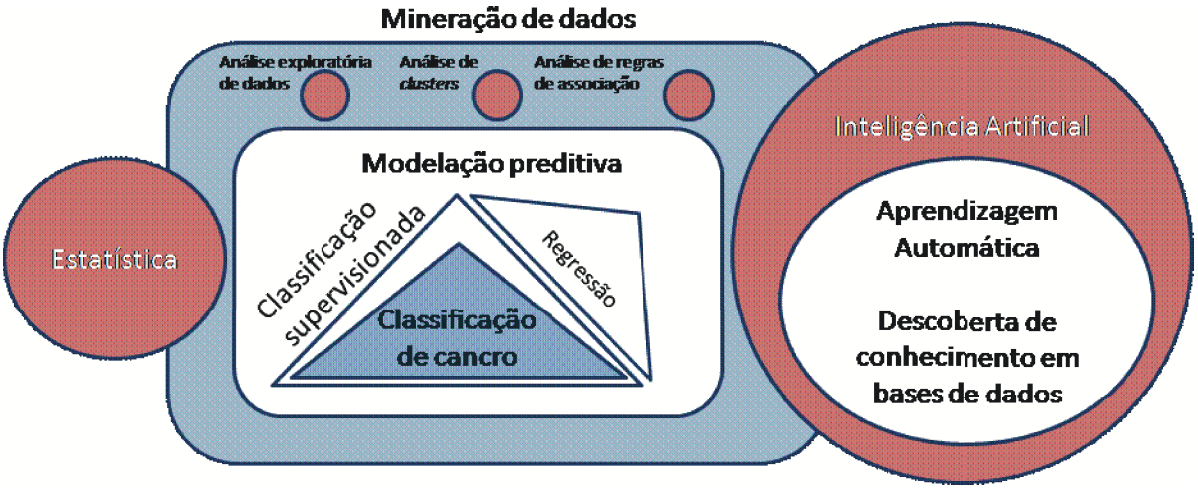


Figura 2.1: Diagrama representativo da mineração de dados como confluência de várias disciplinas. Incidência na classificação supervisionada de cancro.

A descoberta de conhecimento em bases de dados refere-se ao processo não trivial (selecção, pré-processamento, transformação, mineração, avaliação, interpretação) de descoberta de conhecimento útil a partir dos dados [21]. A mineração de dados[3], por outro lado, é uma etapa essencial do processo de DCBD e prende-se com a aplicação de algoritmos computacionais para a extracção de padrões dos dados. Apesar de mineração de dados constituir o passo mais importante no processo de DCBD, usualmente os termos DCBD e mineração de dados são usados como sinónimos.

As técnicas de mineração de dados podem ser subdivididas em quatro tarefas essênciais [47, 54]: *(i)* modelação preditiva, *(ii)* análise de *clusters*, *(iii)* análise de regras de associação e *(iv)* análise exploratória de dados.

O presente trabalho debruça-se sobre a tarefa de classificação supervisionada[4], um tipo de modelo preditivo, que tem como objectivo a construção de um classificador a partir dos dados, *i.e.*, um modelo capaz de prever o valor de uma classe (variável resposta discreta) com base nos valores conhecidos de um conjunto de outras variáveis (variáveis preditivas) [21, 54].

[3]Tradução do termo inglês *Data Mining*.

[4]Nesta dissertação quando for utilizada a palavra classificação é no sentido da classificação supervisionada. A classificação não supervisionada está ligada à análise de *clusters* que não foi estudada neste trabalho.

A aplicação de técnicas de mineração de dados e aprendizagem automática para extrair modelos preditivos a partir de dados de *microarrays* constitui uma área de constante interesse científico devido às suas aplicações imediatas na medicina, particularmente na resolução de diversos problemas relacionados com a classificação de determinados tipos de doenças. Entre estes problemas, nos últimos anos tem sido dada especial atenção ao estudo da classificação de cancro.

Neste capítulo faz-se uma apresentação de todos os conceitos que envolvem a tarefa de classificação de cancro como noção particular da tarefa de classificação. De seguida são apresentados os dois modelos de classificação abordados nesta dissertação, *i.e.* o classificador dos k-vizinhos mais próximos e as máquinas de suporte vectorial. Por fim, o último subcapítulo é destinado à avaliação do desempenho de um modelo de classificação onde é explicitado o método utilizado, ou seja, um método particular de validação cruzada.

2.2 Considerações formais

A secção que se segue é destinada à apresentação dos conceitos básicos relacionados com o problema de classificação. As notações são sobretudo baseadas nas dissertações [8] e [43].

2.2.1 Conceitos básicos

Considere-se $\mathbf{x} = \{x_1, x_2, ..., x_n\}$ um objecto (exemplo, instância), que no caso do problema em estudo são amostras de tecidos, descrito por um conjunto de atributos, $x_1, x_2, ..., x_n$ que correspondem aos genes, $g_1, g_2, ..., g_n$, respectivamente. Cada atributo x_i tem domínio Ω_{x_i} que representa a gama de valores de expressão genética para o gene g_i. Ao conjunto de todas as representações de um objecto, chama-se *espaço de atributos*, representado por $\mathcal{A} = \Omega_{x_1} \times ... \times \Omega_{x_n}$. Este espaço de atributos é, em geral, um subconjunto do espaço Euclidiano n dimensional, assim, $\mathcal{A} \subseteq \mathbb{R}^n$. Assuma-se que o atributo não observado representa o atributo especial classe, onde $\mathcal{C} = \{1, ..., k\}$ é o conjunto de todas as classes envolvidas no problema de classificação e k denota o

número total de classes. Os dados de microarrays figuram assim numa matriz $m \times n$ onde m representa o número de amostras de tecidos e n o número de genes, veja-se a Figura 2.2.

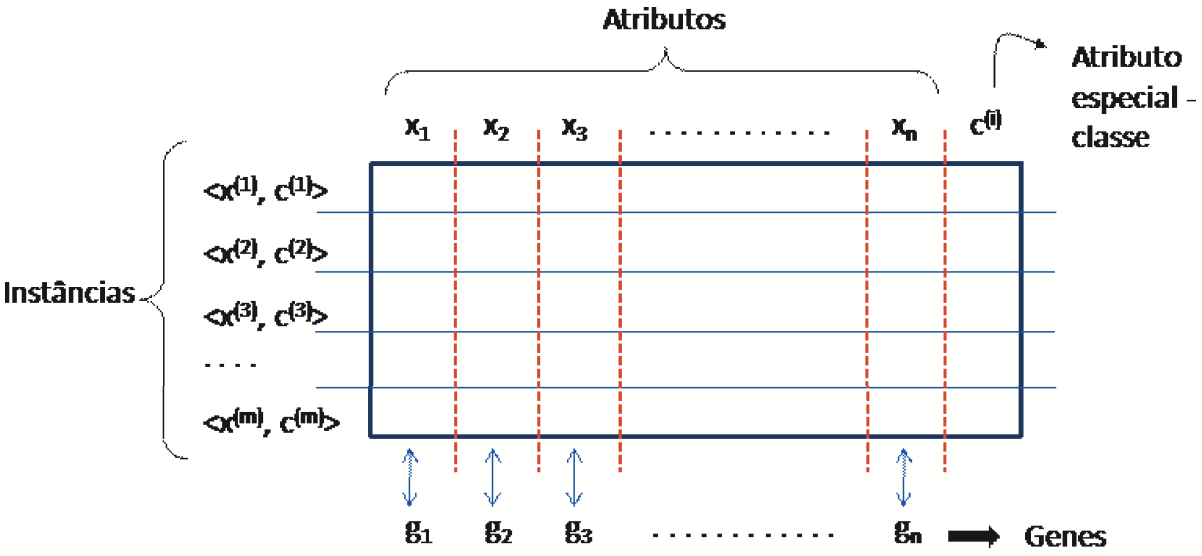


Figura 2.2: Esquema da matriz $m \times n$, onde m representa o número de amostras de tecidos e n o número de genes.

Definição 2.2.1. *Um* ***classificador*** *é uma função* $f : \mathcal{A} \rightarrow \mathcal{C}$ *que atribui uma classe (tipo de cancro)* $c \in \mathcal{C}$*, a cada objecto (amostra de tecido)* $\mathbf{x} \in \mathcal{A}$*, descrito por um conjunto de atributos (genes).*

Seja $\mathcal{D} = \{< \mathbf{x}^{(1)}, c^{(1)} >, < \mathbf{x}^{(2)}, c^{(2)} >, ..., < \mathbf{x}^{(m)}, c^{(m)} >\}$ um conjunto de dados de m tuplos , onde $c^{(i)} \in \mathcal{C}$, com $i = \{1, 2, ..., m\}$, é a classe do tuplo $< \mathbf{x}^{(i)}, c^{(i)} >$.

Definição 2.2.2. *O* ***conjunto de treino*** *é um conjunto de exemplos previamente classificados* $\mathcal{D} = \{< \mathbf{x}^{(1)}, c^{(1)} >, < \mathbf{x}^{(2)}, c^{(2)} >, ..., < \mathbf{x}^{(m)}, c^{(m)} >\}$ *onde cada tuplo* $< \mathbf{x}, c >$*, é composto por um objecto* $\mathbf{x} = \{x_1, x_2, ..., x_n\} \in \mathcal{A}$ *e a sua classe* $c \in \mathcal{C}$*.*

Seja $f : \mathcal{A} \rightarrow \mathcal{C}$ a *função alvo* que é necessário aprender a partir do conjunto de treino $\mathcal{D}$.

Outro conjunto importante na tarefa de classificação, para além do conjunto de treino, é designado por *conjunto de teste*. Os novos exemplos, que não são rotulados *a priori*, fazem parte deste conjunto.

Definição 2.2.3. *A **aprendizagem supervisionada** consiste na tarefa de induzir (aprender) um classificador, ou seja, uma hipótese $h : \mathcal{A} \rightarrow \mathcal{C}$, que melhor aproxime a função alvo a partir do conjunto de treino $\mathcal{D}$.*

Na construção de um classificador existem duas fases fundamentais: a *fase de aprendizagem* (passo indutivo) e a *fase de classificação* (passo dedutivo). Na primeira, existe um conjunto de treino $\mathcal{D}$ com m exemplos previamente classificados onde há a indução de uma hipótese, $h : \mathcal{A} \rightarrow \mathcal{C}$, função determinística, capaz de predizer com alta fiabilidade as classes de futuros exemplos (ainda não rotulados). Na segunda fase, a cada novo exemplo, $\mathbf{x}^{(\text{novo})}$, é atribuída uma classe de acordo com, $c^{(novo)} = h(\mathbf{x}^{(\text{novo})}) \approx f(\mathbf{x}^{(\text{novo})})$.

Antes de se proceder com a secção seguinte convém referir que para cada modelo de aprendizagem, que representa um classificador que foi escolhido para aproximar a função alvo, é necessário seleccionar um algoritmo de aprendizagem. Isto significa que para cada modelo de classificação existem vários algoritmos possíveis.

2.3 Modelos de aprendizagem baseados em instâncias

2.3.1 Introdução

Os modelos de classificação foram desenvolvidos com o propósito de aprender um modelo dos dados que associa uma classe predefinida a um objecto descrito por uma série de atributos. Pode afirmar-se que existem dois passos genéricos que constituem um modelo de classificação:

1. passo indutivo do qual se constrói um modelo dos dados;
2. passo dedutivo para aplicar o modelo construído aos novos exemplos.

Os modelos de classificação que atrasam o processo de modelação dos dados, ou seja, que atrasam o passo indutivo, até o classificador necessitar de classificar as novas instâncias, são designados por modelos de classificação preguiçosos[5]. Outra designação para este tipo de modelos é a de aprendizagem baseada em instâncias. A desvantagem

[5]Tradução do termo inglês *lazy learners*.

imediata deste modelo é o facto de não haver uma construção explícita da estrutura aprendida, o que de certa forma pode contradizer a noção intuitiva de aprendizagem. Um exemplo de um modelo de classificação preguiçoso é designado por *rote learning*. Este modelo memoriza todo o conjunto de treino. De seguida, procede à classificação de um novo objecto se os valores dos seus atributos coincidirem exactamente com os valores dos atributos de pelo menos um dos exemplos de treino, atribuindo à nova instancia a classe desse exemplo. Na verdade, com esta estratégia de classificação é possível que existam exemplos de teste que não coincidam exactamente com os exemplos de treino, e que portanto não serão classificados. No entanto, perante esta situação inconveniente é possível obter uma estratégia de classificação mais flexível que tenha em conta a semelhança entre objectos. Esta estratégia baseia-se no classificador dos k-vizinhos mais próximos. Neste, cada nova instância é comparada com as existentes no conjunto de treino, usando para isso uma métrica adequada, e as k instâncias mais próximas são usadas para classificar o novo objecto. Mesmo assim, como na estratégia anterior, este novo modelo não guarda nenhum conjunto de regras explícitas pois limita-se a armazenar os exemplos. Assim, a fase de classificação envolve um maior custo computacional, pois sempre que chega um novo exemplo para ser classificado é necessário calcular todas as distâncias entre o novo exemplo e cada elemento do conjunto de treino.

2.3.2 O classificador dos k-vizinhos mais próximos

O classificador dos k-vizinhos mais próximos (k-NN[6]) surge da ideia de que objectos que se encontram mais "próximos"no espaço de atributos têm mais possibilidades de pertencerem a uma mesma classe. Desta forma, a tarefa de classificação consiste em, para cada novo objecto, determinar a classe dos objectos mais "próximos"desse e atribuir-lhe a classe predominante. Esta ideia ainda não tem subjacente o valor de k, no entanto, este tem de ser definido. É importante destacar que este é um modelo não paramétrico visto que apenas o valor de k tem de ser definido antes de começar o processo de aprendizagem. Isto é, não há qualquer parâmetro a ser estimado.

Assuma-se que cada objecto corresponde a um vector em $\mathbb{R}^n$, onde n é o número de

[6]Abreviatura do termo inglês k-Nearest Neighbours, k-NN.

atributos. Seja $\mathcal{D} = \{< \mathbf{x}^{(1)}, c^{(1)} >, < \mathbf{x}^{(2)}, c^{(2)} >, ..., < \mathbf{x}^{(m)}, c^{(m)} >\}$ um conjunto de treino com m exemplos previamente rotulados e $\mathbf{z}$ um novo objecto que se pretende classificar.

A "proximidade" mencionada no início desta secção é subjectiva, portanto, para tornar a questão mais objectiva torna-se inevitável definir uma métrica. A mais usada é a distância euclidiana.

Definição 2.3.1. *Considere-se em* $\mathbb{R}^n$ *dois vectores* $x, y \in \mathbb{R}^n$ *onde* $x = (x_1, x_2, ..., x_n)$ *e* $y = (y_1, y_2, ..., y_n)$. *Define-se* ***distância euclidiana****,* d*, como sendo,*
$d = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + ... + (x_n - y_n)^2}$

É de referir que quando todos os atributos são do tipo numérico o cálculo da distância entre dois objectos é directo. Contudo, na presença de pelo menos um atributo nominal é necessário perceber qual a distância entre os diferentes atributos. Para o problema em estudo essa situação não se coloca em virtude dos atributos serem todos numéricos, uma vez que representam valores de expressão genética. Uma outra questão a ter em conta é a desigualdade de importância de alguns atributos em certos problemas, no entanto, para o presente trabalho e aquando da indução deste classificador, todos os atributos têm igual importância.

Posteriormente à Definição 2.3.1 é possível dizer que os k-vizinhos mais próximos do novo exemplo $\mathbf{z}$ referem-se aos k exemplos que estão mais perto usando a distância euclidiana.

A Figura 2.3 ilustra os vizinhos 1,2 e 3 mais próximos do objecto no centro da circunferência. Este objecto vai ser classificado com base na classe mais votada entre os seus

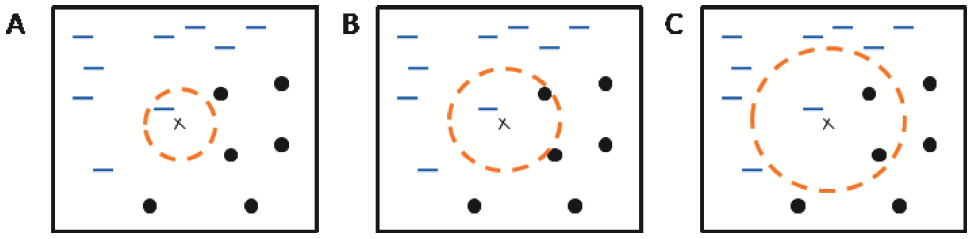


Figura 2.3: Identificação dos k-vizinhos mais próximos de uma instância: **A:** $k = 1$; **B:** $k = 2$; **C:** $k = 3$ (figura adaptada de [47]).

vizinhos. No primeiro caso o exemplo será classificado como sendo da classe "−" uma

vez que foi definido $k = 1$ vizinho. Pelo contrário, na última situação, o número de vizinhos mais próximos é $k = 3$ e dois desses vizinhos pertencem à classe "•". Usando o critério do voto pela maioria, o objecto será classificado como pertencente à classe "•". No caso de haver empate, como acontece na situação do meio, escolhe-se aleatoriamente a classe a atribuir ao objecto. O procedimento clássico para evitar a conjuntura anterior é considerar um número ímpar de vizinhos.

De um modo geral, a escolha do melhor valor para k é um problema a averiguar em cada problema de classificação em concreto. No caso de k ser demasiado pequeno, o classificador dos vizinhos mais próximos pode estar sujeito a um efeito de sobre-ajustamento motivado pelo ruído no conjunto de treino. Por outro lado, se k é demasiado grande pode existir o risco de má classificação de uma instância de teste. Isto acontece porque se o conjunto de vizinhos mais próximos é elevado há a tendência em abranger exemplos que estejam afastados do objecto a classificar. Portanto, ficam dessa forma a ter uma relação de proximidade irreal com a instância a classificar. Alguns estudos empíricos mostram que os melhores resultados obtidos são para $k = 3$ ou $k = 5$ [59], podendo chegar até $k = 10$.

Sejam $\mathbf{z} =< \mathbf{x}', c' >$ um exemplo de teste, $< \mathbf{x}, c > \in \mathcal{D}$ um exemplo de treino genérico pertencente ao conjunto de treino $\mathcal{D}$ e ainda $\mathcal{D}_z$ a lista de vizinhos mais próximos do exemplo de teste $\mathbf{z}$.

Algoritmo 1

Seja k o número de vizinhos mais próximos e $\mathcal{D}$ o conjunto de exemplos de treino.

`para cada` exemplo de teste $z =< x', c' >$ `fazer`

Calcular $d(\mathbf{x}', \mathbf{x})$, a distância entre z e todo o exemplo $(\mathbf{x}, c) \in \mathcal{D}$

Seleccionar $\mathcal{D}_z \subseteq \mathcal{D}$, o conjunto dos k exemplos de treino mais próximos de z

$c' = \text{argmax}_v \sum_{<\mathbf{x}^{(i)}, c^{(i)}> \in \mathcal{D}_z, i \in \{1,\ldots,m\}} I(v = c^{(i)})$

`terminar`

A penúltima linha do Algoritmo 1 [47] tem como objectivo classificar o exemplo de teste, a partir da lista de vizinhos mais próximos desse exemplo, tendo em conta o critério do voto pela maioria. Nesse critério, v representa uma classe genérica do conjunto de classes predefinidas do problema de classificação e $c^{(i)}$ é a classe de um dos exemplos pertencentes ao conjunto dos vizinhos mais próximos de $\mathbf{z}$, ou seja, $\mathcal{D}_z$.

A função indicatriz $I(\cdot)$ retorna o valor um se o seu argumento for uma proposição verdadeira e o valor zero caso contrário. Tendo em conta que em certos problemas nem todos os atributos têm o mesmo impacto, é possível reescrever o modelo de votação anterior através da introdução de um factor multiplicativo. Este caso sai do âmbito deste trabalho.

2.4 Máquinas de suporte vectorial

2.4.1 Introdução

Os modelos de aprendizagem que pertencem ao grupo dos *lazy learners* não requerem a construção de um modelo, como é referido na Secção 2.3. Nestes, o custo de classificar um exemplo de teste é elevado e provém da necessidade de calcular as medidas de proximidade entre todas as instâncias de treino e a de teste. Em contrapartida, os *eager learners* despendem grande parte dos recursos de computação na construção do modelo de aprendizagem. Contudo, quando um exemplo de teste está na condição de lhe ser atribuída uma classe, o procedimento tende a ser mais imediato.

Um caso de um modelo de aprendizagem automática que se inclui na classe dos *eager learners* e que tem recebido muita atenção no meio científico é o modelo das máquinas de suporte vectorial (MSV)[7]. Este foi originalmente introduzido por Vapnik e os seus colaboradores na década de 90 [66]. A ideia a ele subjacente, considerando o problema mais básico de apenas duas classes, é a construção de um hiperplano[8] de margem máxima que separa objectos pertencentes a classes diferentes. Os conceitos básicos deste modelo encontram-se definidos na literatura especializada, por exemplo em [66], como também em [6, 46, 47, 54, 74].

Este modelo tem sido aplicado a problemas de classificação em diferentes áreas e em particular na área da genética, tendo-se assim tornando muito popular. Mais ainda,

[7] Tradução do termo inglês *Support Vector Machines*, em abreviatura SVM.

[8] Um hiperplano é uma extensão dos conceitos de duas e três dimensões, rectas e planos respectivamente, a dimensões superiores. É de referir que num espaço de dimensão n, um hiperplano é um objecto de dimensão $n-1$, da mesma forma que uma recta é um objecto de dimensão um num espaço de dimensão dois.

tem sido bem sucedido nas suas diversas aplicações e é implementado pelo facto de ter uma sólida fundamentação teórica.

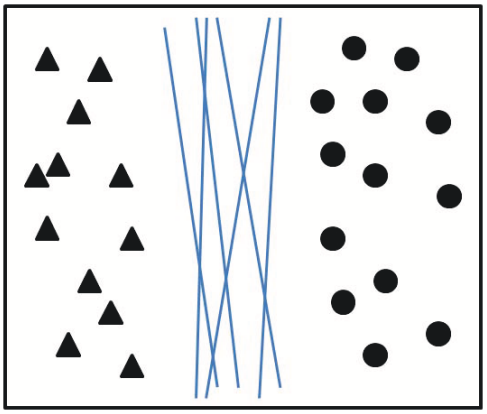


Figura 2.4: Representação de possíveis fronteiras de decisão para um conjunto de dados linearmente separável. Classes representadas por círculos e triângulos (figura adaptada de [47]).

Antes de se proceder com as considerações formais acerca deste classificador, observe-se a Figura 2.4 onde é possível visualizar dados linearmente separáveis. Embora existam infinitos hiperplanos que separam correctamente as duas classes, deve ser escolhido aquele hiperplano que melhor generalize o conjunto de treino, ou seja, aquele que classifique com maior fiabilidade futuros exemplos (não rotulados).

De um modo esquemático, a Figura 2.5 pretende ilustrar as diferenças entre dois hiperplanos que separam correctamente objectos pertencentes a classes distintas. Nessa figura são visíveis dois tipos de objectos e duas fronteiras de decisão ($\mathcal{F}_1$ e $\mathcal{F}_2$) diferentes que separam correctamente todos os objectos. Cada fronteira de decisão é acompanhada por dois hiperplanos, *i.e.*, os hiperplanos h_{11} e h_{12} associados a $\mathcal{F}_1$ e os hiperplanos h_{21} e h_{22} associados a $\mathcal{F}_2$. Ambos os pares de hiperplanos são obtidos fazendo uma translação da fronteira de decisão respectiva até ao objecto mais próximo de cada classe. A distância entre ambos os hiperplanos é chamada de *margem do classificador*. É possível constatar visualmente, que a margem para $\mathcal{F}_1$ é consideravelmente superior do que para $\mathcal{F}_2$. Como consequência, conclui-se que para este exemplo, $\mathcal{F}_1$ é o hiperplano de margem máxima para as instâncias observadas.

Os exemplos localizados em posições mais afastadas do hiperplano de margem máxima não entram na sua especificação. Pelo contrário, as instâncias que estão localizadas

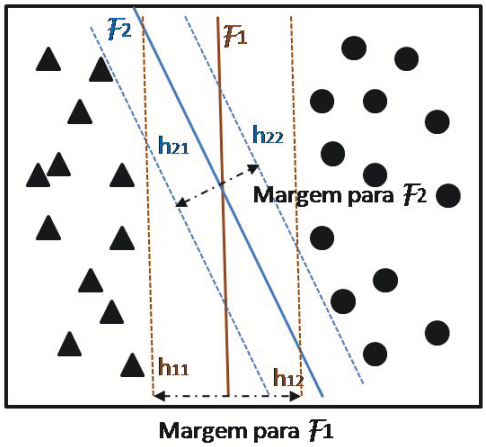


Figura 2.5: Representação a duas dimensões da margem de uma fronteira de decisão linear que separa um conjunto de dados com duas classes distintas, círculos e triângulos (figura adaptada de [47]).

mais próximas do hiperplano, denominadas *vectores de suporte*, são essenciais para precisar a sua expressão matemática. Este tipo de exemplos deverá ser uma pequena percentagem sobre o conjunto total de exemplos, todavia, deve existir pelo menos um vector de suporte por cada classe. Será conveniente referir que o conjunto de vectores de suporte define por si só o hiperplano óptimo para um dado problema, pelo que os restantes exemplos são irrelevantes. Estes últimos podem mesmo ser eliminados sem alterar a posição e orientação do hiperplano de margem máxima, o mesmo já não acontece com os vectores de suporte [54].

2.4.2 Caso linear: separável

Considere-se o problema binário onde são dados m tuplos, $\{< \mathbf{x}^{(1)}, c^{(1)} >, < \mathbf{x}^{(2)}, c^{(2)} > , ..., < \mathbf{x}^{(m)}, c^{(m)} >\}$. Este conjunto é designado por conjunto de treino. O vector $\mathbf{x}^{(i)}$ corresponde aos níveis de expressão genética para n genes e a classe $c^{(i)}$ tem apenas dois valores possíveis (± 1), para $i \in \{1, 2, ...m\}$. O objectivo é assim aprender uma função multivariada a partir do conjunto de treino que determine, com precisão, a classe de um novo exemplo, $f(\mathbf{x}^{novo}) = c^{novo}$.

Suponha-se uma função de classificação linear

$$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b \tag{2.1}$$

onde $\mathbf{w}$ e b são parâmetros do modelo.

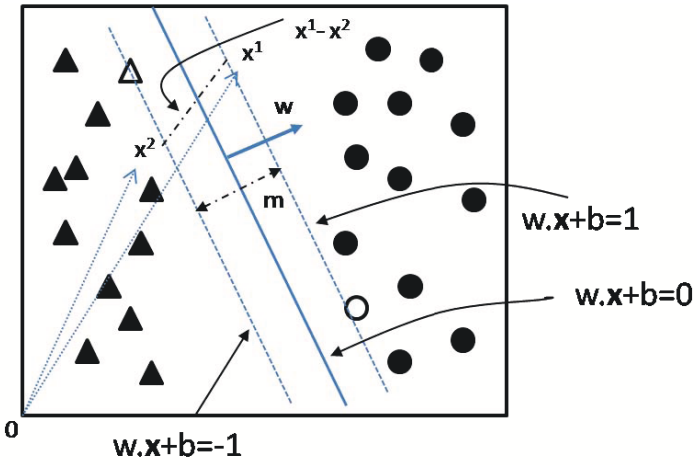


Figura 2.6: Fronteira de decisão e margem de uma MSV. O conjunto de dados é constituído por duas classes representadas por círculos e triângulos (figura adaptada de [47]).

A Figura 2.6 mostra um exemplo a duas dimensões de um conjunto de treino que consiste em objectos de duas classes diferentes. A linha sólida representa a fronteira de decisão que separa os dois tipos de objectos. Refira-se que um qualquer objecto $\mathbf{a}$, que esteja representado na fronteira de decisão, satisfaz a equação $\mathbf{w} \cdot \mathbf{a} + b = 0$.

Para qualquer círculo, $\mathbf{x}_c$, situado acima da fronteira de decisão tem-se

$$\mathbf{w} \cdot \mathbf{x}_c + b = k \text{ , onde } k > 0 \tag{2.2}$$

Da mesma forma, para qualquer triângulo, $\mathbf{x}_t$, localizado abaixo da fronteira de decisão tem-se

$$\mathbf{w} \cdot \mathbf{x}_t + b = k' \text{ , onde } k' < 0 \tag{2.3}$$

Se todos os círculos forem atribuídos à classe "+1"e todos os triângulos à classe "−1", é possível então predizer a classe c de qualquer exemplo $\mathbf{z}$ da seguinte forma

$$c = \begin{cases} 1 & \text{se } \mathbf{w} \cdot \mathbf{z} + b > 0 \\ -1 & \text{se } \mathbf{w} \cdot \mathbf{z} + b < 0 \end{cases} \tag{2.4}$$

Fazendo uma transformação dos parâmetros $\mathbf{w}$ e b da fronteira de decisão, e considerando um vector de atributos $\mathbf{x}$ genérico, é possível escrever os hiperplanos paralelos

b_1 e b_2 da seguinte forma

$$b_1 : \mathbf{w} \cdot \mathbf{x} + b = 1 \tag{2.5}$$

$$b_2 : \mathbf{w} \cdot \mathbf{x} + b = -1 \tag{2.6}$$

A margem do classificador foi definida previamente como sendo a distância entre os dois hiperplanos que definem a fronteira de decisão. Na Figura 2.6 a margem está representada com a letra m, e pode ser calculada usando os pontos x^1 localizado em b_1e x^2 localizado em b_2. Ao substituir os pontos x^1 e x^2 nas Equações 2.5 e 2.6 respectivamente, e subtraindo-as, obtém-se

$$\mathbf{w} \cdot (\mathbf{x}^1 - \mathbf{x}^2) = 2$$

Pela definição de coseno de um ângulo, tem-se

$$\cos(\mathbf{w}, \mathbf{x}^1 - \mathbf{x}^2) = \frac{\mathbf{w} \cdot (\mathbf{x}^1 - \mathbf{x}^2)}{\|\mathbf{w}\|\|\mathbf{x}^1 - \mathbf{x}^2\|}$$

ao mesmo tempo, é perceptível na Figura 2.6 que

$$\cos(\mathbf{w}, \mathbf{x}^1 - \mathbf{x}^2) = \frac{m}{||\mathbf{x}^1 - \mathbf{x}^2||}$$

Assim, a margem m é dada por

$$\begin{aligned} \frac{\mathbf{w} \cdot (\mathbf{x}^1 - \mathbf{x}^2)}{\|\mathbf{w}\|\|\mathbf{x}^1 - \mathbf{x}^2\|} &= \frac{m}{\|\mathbf{x}^1 - \mathbf{x}^2\|} \\ \mathbf{w} \cdot (\mathbf{x}^1 - \mathbf{x}^2) &= m\|\mathbf{w}\| \\ \therefore m &= \frac{2}{\|\mathbf{w}\|} \end{aligned}$$

A fase de aprendizagem do classificador MSV abrange a estimação dos parâmetros $\mathbf{w}$ e b associados à fronteira de decisão. Desta forma, estes dois parâmetros devem ser escolhidos de tal forma que

$$\mathbf{w} \cdot \mathbf{x} + b \geq 1 \text{ se } c = 1 \tag{2.7}$$

$$\mathbf{w} \cdot \mathbf{x} + b \leq -1 \text{ se } c = -1 \tag{2.8}$$

Estas duas desigualdades impõem que todos os objectos pertencentes à classe $c = 1$, círculos, estejam localizados acima do hiperplano $\mathbf{w} \cdot \mathbf{x} + b = 1$, enquanto que os exemplos pertencentes à classe $c = -1$, triângulos, estejam localizados abaixo do hiperplano $\mathbf{w} \cdot \mathbf{x} + b = -1$. Outra forma de escrever as duas desigualdades anteriores é usar uma única desigualdade equivalente

$$c^{(i)}(\mathbf{w} \cdot \mathbf{x}^{(i)} + b) \geq 1 \text{ , para } i = 1, 2, ..., m \tag{2.9}$$

Para além desta imposição é ainda requerido que a margem da fronteira de decisão seja máxima. Neste caso o classificador MSV encontra um hiperplano de margem máxima, ou seja, um hiperplano que maximize a distância entre este e os pontos mais próximos do conjunto de dados, sujeito à restrição anterior, *i.e.*, pontos pertencentes a classes diferentes permaneçam em lados opostos do hiperplano. O seguinte problema de optimização satisfaz o objectivo pretendido

$$\begin{aligned} &\max_{\mathbf{w}} \min_{\mathbf{x}^{(i)}} \frac{c^{(i)}(\mathbf{w} \cdot \mathbf{x}^{(i)} + b)}{\|\mathbf{w}\|} \\ &\quad \text{s.a. } c^{(i)}(\mathbf{w} \cdot \mathbf{x}^{(i)} + b) \geq 1 \text{ para todo } \mathbf{x}^{(i)}\, i \in \{1, 2, ..., m\} \end{aligned}$$

No entanto, uma formulação mais simples e equivalente [66] do problema anterior é a minimização da seguinte função objectivo

$$f(\mathbf{w}) = \frac{||\mathbf{w}||^2}{2}$$

Desta forma, a tarefa de aprendizagem do classificador MSV pode ser formulada pelo seguinte problema de optimização

$$\begin{aligned} &\min_{\mathbf{w}} \frac{||\mathbf{w}||^2}{2} \\ &\quad \text{s.a. } c^{(i)}(\mathbf{w} \cdot \mathbf{x}^{(i)} + b) \geq 1 \text{ , para } i = 1, 2, ..., m \end{aligned} \tag{2.10}$$

A função objectivo é quadrática e as restrições do problema com respeito aos parâmetros $\mathbf{w}$ e b são lineares. Este problema de optimização é classificado como um problema

de optimização convexa[9]. Para proceder à sua resolução pode usar-se o método dos multiplicadores de Lagrange [47].

Primeiro é necessário reescrever a função objectivo de modo a ter em conta as restrições do Problema 2.10. A nova função objectivo é conhecida como o Lagrangeano para o problema de optimização:

$$\mathcal{L}_P = \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{m} \lambda_i \Big(c^{(i)}(\mathbf{w} \cdot \mathbf{x}^{(i)} + b) - 1 \Big), \tag{2.11}$$

onde os parâmetros λ_i são designados por multiplicadores de Lagrange.

Para minimizar o Lagrangeano calculam-se as derivadas parciais de $\mathcal{L}_P$ em relação aos parâmetros $\mathbf{w}$ e b e igualam-se a zero, assim tem-se

$$\frac{\partial \mathcal{L}_P}{\partial \mathbf{w}} = 0 \Longrightarrow \mathbf{w} = \sum_{i=1}^{m} \lambda_i c^{(i)} \mathbf{x}^{(i)} \tag{2.12}$$

$$\frac{\partial \mathcal{L}_P}{\partial b} = 0 \Longrightarrow \sum_{i=1}^{m} \lambda_i y_i = 0 \tag{2.13}$$

A partir das Equações 2.12 e 2.13 ainda não é possível deduzir os parâmetros $\mathbf{w}$ e b, uma vez que os multiplicadores de Lagrange são desconhecidos. Para abordar o facto das restrições do Problema 2.10 serem desigualdades torna-se necessário passá-las a restrições de igualdade. Esta alteração é praticável apenas se os multiplicadores de Lagrange tomam valores não negativos. Tal transformação conduz às seguintes restrições dos multiplicadores de Lagrange, conhecidas por condições de Karush-Kuhn-Tucker (KKT) [35, 37]:

$$\lambda_i \geq 0 \tag{2.14}$$

$$\lambda_i [c^{(i)}(\mathbf{w} \cdot \mathbf{x}^{(i)} + b) - 1] = 0 \tag{2.15}$$

[9]Considere-se o seguinte problema de optimização,

$$\begin{aligned} &\min f(x) \\ &\quad \text{s.a. } g_i(x) \leq 0 \text{ para todo } i = 1, 2, ..., m \\ &\quad \text{s.a. } h_i(x) = 0 \text{ para todo } i = 1, 2, ..., p \end{aligned}$$

onde $f(x) : \mathbb{R}^n \to \mathbb{R}$ é uma função convexa, as desigualdades $g_i(x)$ são funções convexas e as restrições $h_i(x)$ são funções afim. Este problema define-se como um problema de optimização convexa.

A maioria dos multiplicadores de Lagrange toma o valor zero após a aplicação da restrição dada pela Equação 2.15. Repare-se que a restrição indica que o multiplicador de Lagrange λ_i tem de tomar o valor zero a não ser que a instância $\mathbf{x}^{(i)}$ satisfaça a equação $c^{(i)}(\mathbf{w} \cdot \mathbf{x}^{(i)} + b) = 1$. Tal exemplo de treino, com $\lambda_i > 0$, situa-se num dos hiperplanos que estão de cada lado da fronteira de decisão, e chama-se, o já acima referido, vector de suporte. As instâncias de treino que não se situam nos hiperplanos têm associados multiplicadores de Lagrange $\lambda_i = 0$. As Equações 2.12 e 2.15 sugerem que os parâmetros $\mathbf{w}$ e b, que definem a fronteira de decisão, dependem apenas dos vectores de suporte. Como referido em [47], mesmo após estas transformações o processo de optimização continua complexo, pois envolve uma grande quantidade de parâmetros: $\mathbf{w}$, b e λ_i's. O problema pode ser simplificado através da transformação do Lagrangeano primal, $\mathcal{L}_P$, no Lagrangeano dual, $\mathcal{L}_D$, que é apenas função dos multiplicadores de Lagrange e é dado por,

$$\mathcal{L}_D = \sum_{i=1}^{m} \lambda_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \lambda_i \lambda_j c^{(i)} c^{(j)} \mathbf{x}^{(i)} \cdot \mathbf{x}^{(j)} \tag{2.16}$$

Uma vez determinados os valores dos multiplicadores λ_i, através da maximização do problema dual, calcula-se a solução admissível para os parâmetros $\mathbf{w}$ e b com base nas Equações 2.12 e 2.15. A fronteira de decisão pode ser assim expressa da seguinte forma

$$\Big(\sum_{i=1}^{m} \lambda_i c^{(i)} \mathbf{x}^{(i)} \Big) + b = 0 \tag{2.17}$$

onde b é obtido resolvendo a Equação 2.15.

A forma como o classificador MSV foi desenvolvido permite-lhe seleccionar um hiperplano entre os vários possíveis, no entanto, nem sempre é factível encontrar tal fronteira entre as classes. Em certas situações acontecem más classificações de exemplos [6, 47] e, como consequência, o problema torna-se não linearmente separável. Mesmo assim é possível encontrar uma fronteira de decisão linear. A solução passa por usar uma margem suave que aceite exemplos de treino mal classificados. Isto pode ser alcançado introduzindo variáveis de desvio na função objectivo para permitir que as restrições possam ser violadas, levando à modificação da formulação do problema. Neste caso, tem de ocorrer um compromisso entre encontrar um hiperplano com margem máxima,

minimizando $||\mathbf{w}||$, e encontrar um hiperplano que separe efectivamente os exemplos das duas classes, minimizando as variáveis de desvio [46].

2.4.3 Caso não linear

A maior parte dos problemas da vida real estão associados a problemas onde os dados não são linearmente separáveis. Por outras palavras, apresentam fronteiras de decisão não lineares, e portanto, o caso da secção anterior torna-se demasiado elementar para aplicações práticas.

A solução para este problema baseia-se na transformação dos dados do seu sistema de coordenadas original para um novo sistema onde uma fronteira de decisão linear possa ser definida. Esta fronteira linear é usada para separar as instâncias no espaço transformado, designado por *espaço de características*, $\Phi(\mathbf{x})$. Posteriormente a este passo é possível aplicar a metodologia apresentada na Secção 2.4.2. É de destacar que mapear o conjunto de treino num espaço de dimensionalidade superior acarreta custos ao nível computacional e teórico.

A formulação do problema não linear é muito semelhante ao caso linear, veja-se

$$\begin{aligned} &\min_{\mathbf{w}} \frac{||\mathbf{w}||^2}{2} \\ &\text{s.a. } c^{(i)}(\mathbf{w} \cdot \Phi(\mathbf{x}^{(i)}) + b) \geq 1 \text{ , para } i = 1, 2, ...m \end{aligned} \tag{2.18}$$

A maior diferença reside na substituição dos valores dos atributos, $\mathbf{x}^{(i)}$, por uma função não linear dos mesmos, $\Phi(\mathbf{x}^{(i)})$. Os operadores são semelhantes, contudo, neste caso é necessário fazer cálculos usando os atributos no novo espaço e estes podem tornar-se difíceis. Esta dificuldade justifica-se pelo facto de ser complexo encontrar a formulação explícita da função que mapeia o sistema original de coordenadas no novo espaço. Para além disso, é preciso assegurar ao mesmo tempo que uma fronteira de decisão linear, $\mathbf{w} \cdot \Phi(\mathbf{x}) + b = 0$, possa ser construída no espaço de características. De forma semelhante à abordagem da secção anterior é possível construir o Lagrangeano dual para o Problema de optimização com restrições 2.18:

$$\mathcal{L}_D = \sum_{i=1}^{m} \lambda_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \lambda_i \lambda_j c^{(i)} c^{(j)} \Phi(\mathbf{x}^{(i)}) \cdot \Phi(\mathbf{x}^{(j)}) \tag{2.19}$$

Uma vez determinados os multiplicadores λ_i's, os parâmetros $\mathbf{w}$ e b podem ser derivados das seguintes equações

$$\mathbf{w} = \sum_{i=1}^{m} \lambda_i c^{(i)} \Phi(\mathbf{x}^{(i)}) \tag{2.20}$$

$$\lambda_i \Big[c^{(i)} \big(\sum_{j=1}^{m} \lambda_j c^{(j)} \Phi(\mathbf{x}^{(j)}) \cdot \Phi(\mathbf{x}^{(i)}) + b \big) - 1 \Big] = 0 \tag{2.21}$$

que são análogas às Equações 2.12 e 2.13 da Secção 2.4.2.

Deste modo, para classificar uma instância de teste, $\mathbf{z}$, esta pode ser classificada usando a seguinte função

$$f(\mathbf{z}) = sign(\mathbf{w} \cdot \Phi(\mathbf{z}) + b) = sign\Big(\sum_{i=1}^{m} \lambda_i c^{(i)} \Phi(\mathbf{x}^{(i)}) \cdot \Phi(\mathbf{z}) + b \Big) \tag{2.22}$$

A função 2.22 envolve cálculos com o produto interno entre pares de vectores no espaço transformado, $\Phi(\mathbf{x}^{(i)}) \cdot \Phi(\mathbf{z})$. Dada a dificuldade desse problema, a resolução passa pela utilização de funções *kernel* [47].

As funções *kernel* são funções que são calculadas no espaço original de atributos e fazem o papel de produto interno no espaço de dimensão superior. Este tipo de funções permite resolver todo o problema sem nunca ter de representar explicitamente a função que mapeia o espaço original de atributos no novo espaço. Isto acontece porque as funções *kernel*, usadas em problemas de MSV no caso não linear, precisam de satisfazer um princípio conhecido como Teorema de Mercer [66], ou seja, serem definidas positivas.

Teorema 2.4.1 (**Mercer**). *Uma função* kernel *K pode ser expressa como,*

$$K(u, v) = \Phi(u) \cdot \Phi(v)$$

se e só se, para qualquer função $g(\mathbf{x})$ tal que $\int g(\mathbf{x})^2 d\mathbf{x}$ seja finita, então

$$\int K(\mathbf{x}, \mathbf{y}) g(\mathbf{x}) g(\mathbf{y}) d\mathbf{x}\, d\mathbf{y} \geq 0$$

Este teorema assegura que as funções *kernel* possam sempre ser escritas como o produto interno entre dois vectores num espaço de dimensão superior, contudo, são calculadas no espaço original de atributos [47].

A selecção da função *kernel* é importante pois é ela que vai estabelecer a precisão do classificador. Usualmente esta selecção pressupõe um conhecimento prévio ou esperado do comportamento dos dados [6].

Uma instância de teste, $\mathbf{z}$, é assim classificada de acordo com a seguinte equação:

$$\begin{aligned} f(\mathbf{z}) &= sign\Big(\sum_{i=1}^{m} \lambda_i c^{(i)} \Phi(\mathbf{x}^{(i)}) \cdot \Phi(\mathbf{z}) + b\Big) \\ &= sign\Big(\sum_{i=1}^{m} \lambda_i c^{(i)} K(\mathbf{x}^{(i)}, \mathbf{z}) + b\Big) \end{aligned} \tag{2.23}$$

2.4.4 Classificação multi-classe

Nesta subsecção é abordado o tema de classificação multi-classe de forma sumária. O objectivo é apresentar de modo sucinto este tema, uma vez que é usada a nível computacional uma das estratégias aqui referidas. Como este tema não é o propósito deste trabalho, adopta-se uma abordagem simplificada que foi baseada no trabalho [46] o qual poderá ser consultado para mais esclarecimentos.

Existem vários problemas binários para os quais se pode aplicar directamente a teoria apresentada nas secções anteriores. É necessário, no entanto, recorrer à classificação multi-classe quando o problema apresenta mais do que duas classes. Torna-se assim claro que esta classificação é mais complexa que a predição binária, porque o algoritmo seleccionado para o modelo em causa tem de aprender a construir um maior número de fronteiras de separação. Na classificação binária o algoritmo eleito consegue discriminar apenas uma das classes com uma apropriada fronteira de decisão, pois a outra é apenas a complementar. Nos problemas de classificação com mais de duas classes, cada uma delas tem de ser definida explicitamente. Assim, o problema inicial é decomposto em vários subproblemas binários, combinando-se no final as predições de todos estes.

Esta decomposição baseia-se na estratégia de *dividir para conquistar*. Neste caso, o problema inicial é decomposto em vários subproblemas binários mais fáceis de resolver. Para estes, existe uma panóplia de algoritmos que podem ser usados. Para além da escolha da decomposição do problema inicial e do classificador, é preciso ainda determinar a estratégia para combinar os classificadores binários e originar a predição final. A título de exemplo, informação sobre este tópico pode ser consultada em [31].

O *output coding* é utilizado para resolver o problema da combinação dos classificadores binários [16]. Esta estratégia baseia-se na ideia de que havendo k classificadores treinados nas várias partições do conjunto de dados, o novo exemplo a classificar é transformado num vector de saída k dimensional. Qualquer elemento do vector de saída é o resultado da classificação de cada um dos k classificadores. Um *codebook* é usado para associar o resultado do vector a uma das classes predefinidas. Existem duas versões para a estratégia de *output coding*: *one-versus-all* (OVA) e *all-pairs* (AP). Na primeira versão, dadas k classes, k classificadores independentes são construídos. Nesta construção, o i-ésimo classificador é treinado para separar os exemplos que pertencem à classe i de todas as outras classes. O *codebook* é uma matriz diagonal e a predição final é baseada no classificador que produz a mais elevada medida de certeza

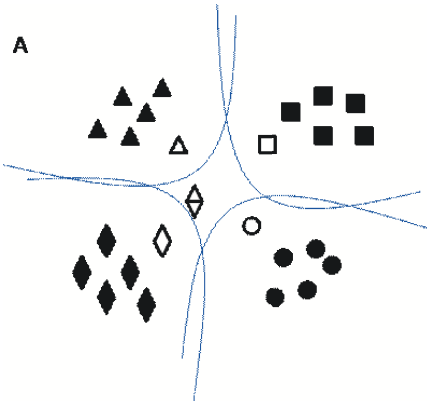
$$class = \text{argmax}_{i=1\ldots k} f_i$$

onde f_i é medida de certeza do i-ésimo classificador.

Ir-se-á agora explicitar um exemplo da estratégia OVA, em virtude de ser a opção utilizada neste estudo.

Na Figura 2.7 podem-se observar quatro classes diferentes com as respectivas fronteiras de decisão (Figura 2.7 A) e os quatro classificadores (Figura 2.7 B). Os quatros classificadores são treinados. O primeiro classificador discrimina a classe ▲ de todas as outras classes; o segundo classificador discrimina a classe ■ de todas as outras classes. O mesmo padrão verifica-se para as classes • e ⧫. Neste exemplo as instâncias de teste são representadas pelos símbolos: $\triangle, \square, \diamond, \circ, \diamond$. O *codebook* (Figura 2.7 B) representa a parte superior da tabela. Nessa parte estão representados os resultados óptimos dos classificadores. O classificador $C(\blacktriangle)$ que aprendeu a discriminar a classe ▲ das restantes, no caso óptimo coloca +1 para os exemplos classificados com ▲ e 0 para o caso contrário. Na parte inferior da tabela pode-se consultar os resultados das predições dos classificadores para os exemplos de teste dos quais se desconhece a classe. O classificador $C(\bullet)$ mostra um valor de certeza de 0.90 para o exemplo de teste "$\circ$"e consequentemente é classificado como pertencendo à classe •.

Na implementação computacional dos algoritmos de aprendizagem para máquinas de suporte vectorial usados no contexto do presente trabalho, foi usada a estratégia OVA para o caso de problemas com mais de duas classes.



B

		Classificador C(▲)	C(■)	C(●)	C(⧫)	
classes	▲	+1	0	0	0	
	■	0	+1	0	0	
	●	0	0	+1	0	Classe
	⧫	0	0	0	+1	predita
teste	△	**0.80**	0.20	0.01	0.20	▲
	◊	0.50	**0.60**	0.30	0.10	⧫
	○	0.10	0.10	**0.90**	0.20	●
	□	0.30	0.10	0.40	**0.80**	■
	◊	0.60	**0.70**	0.10	0.10	⧫

Figura 2.7: Classificação OVA. **A.** Quatro fronteiras de decisão. **B.** O *codebook* para classificação OVA (figura extraída de [46]).

2.5 Avaliação do desempenho de um classificador

Uma vez induzido um classificador a partir de um conjunto de dados é preciso avaliar a sua capacidade de generalização, ou seja, a capacidade de classificar correctamente futuros exemplos (que não foram usados para construir o classificador).

A avaliação do desempenho de um classificador é facilmente exequível quando existe um grande volume de dados (elevado número de instâncias) disponíveis. Todavia, apesar das bases de dados usadas na área da genética envolverem dados de elevada dimensão, frequentemente estas estão associados a um número de instâncias reduzido. Este facto dificulta a determinação do erro verdadeiro associado a um classificador.

Definição 2.5.1. *O* ***erro verdadeiro****, Err_v, de um classificador h é a probabilidade de classificar erroneamente um exemplo seleccionado aleatoriamente,* i.e.*,*

$$Err_v(h) = P_{\mathbf{x}\in\mathcal{A}}[h(\mathbf{x}) \neq c]\,, < \mathbf{x}, c >: \mathbf{x} \in \mathcal{A}, c = f(\mathbf{x})$$

Seria possível determinar o erro verdadeiro se fosse factível dispor de um conjunto de dados ilimitado. Perante a inviabilidade desta situação é necessário recorrer à estimação do erro verdadeiro usando um conjunto limitado de dados, sendo a medida natural do desempenho de um classificador a sua *taxa de erro*. A taxa de erro é assim definida como sendo a proporção de exemplos incorrectamente classificados. Antes de se definir a taxa de erro, defina-se a função de perda 0-1,

Definição 2.5.2. *A "**função de perda** $0-1$", $\delta(\cdot,\cdot,\cdot)$, de um classificador h induzido a partir do conjunto de dados $\mathcal{D}$ define-se como sendo:*

$$\delta(\mathbf{x}, f(\mathbf{x}), h(\mathbf{x})) = \begin{cases} 1 & , \ se\ f(\mathbf{x}) \neq h(\mathbf{x}) \\ 0 & , \ se\ f(\mathbf{x}) = h(\mathbf{x}) \end{cases} \qquad \mathbf{x} \in \mathcal{A}$$

A função $0-1$ mede a "perda", ou custo numérico, de atribuir a um exemplo $\mathbf{x}$ a classe $c' = h(\mathbf{x})$ (classe predita) quando a sua classe verdadeira é $c = f(\mathbf{x})$. Isto significa, que se a classificação do exemplo é correcta a classe predita é igual à classe verdadeira, por isso não existe perda, ou seja, a função de perda toma o valor 0, caso contrário, toma o valor 1. A definição de aprendizagem supervisionada pode ser agora reformulada por:

*Dado um conjunto de dados $\mathcal{D}$ com m exemplos previamente classificados, a tarefa de **aprendizagem supervisionada** consiste em induzir um classificador $h : \mathcal{A} \rightarrow \mathcal{C}$ que minimize a função de perda 0-1.*

Definição 2.5.3. *A **taxa de erro**, TE, de um classificador h com relação a um conjunto de dados $\mathcal{D}$ é definida como a proporção dos exemplos incorrectamente classificados, ou seja,*

$$TE(h(\mathbf{x}), \mathcal{D}) = \frac{1}{m} \sum_{<\mathbf{x}^{(i)}, c^{(i)}> \in \mathcal{D}, i \in \{1,2,\ldots,m\}} \delta(\mathbf{x}^{(i)}, c^{(i)}, h(\mathbf{x}^{(i)}))$$

Para além da taxa de erro, é ainda utilizada frequentemente a *taxa de acertos*, que pelo contrário, é a proporção de exemplos correctamente classificados.

Perante a necessidade de avaliar com credibilidade um classificador, o desempenho do mesmo não pode ser avaliado com base numa medida de erro calculada em função dos mesmos exemplos usados no processo de aprendizagem [54]. Isto é, deve-se avaliar o desempenho do classificador naqueles exemplos que não foram usados para a sua indução. A ideia básica consiste em dividir o conjunto de dados disponível em dois subconjuntos: *(i)* um conjunto de treino usado pelo algoritmo de aprendizagem para induzir o classificador e *(ii)* um conjunto de teste, retirado de modo independente da mesma população de exemplos possíveis e que serão usados para estimar a taxa de erro.

A taxa de erro calculada sobre o conjunto de treino não é um bom indicador do desempenho futuro de um classificador. Isto acontece porque qualquer estimativa de desempenho baseado nos dados de treino será demasiado optimista [74]. O classificador induzido pode sobre-ajustar o conjunto de treino, o que significa que o modelo fica mais ajustado às instâncias usadas para a sua indução do que àquelas que futuramente vão ser classificadas.

Existem vários métodos para estimar a taxa de erro baseados em diferentes partições do conjunto de dados em conjunto de treino e teste. Na próxima secção apresenta-se o método utilizado neste trabalho.

2.5.1 Validação cruzada

No caso mais simples é possível dividir o conjunto inicial de dados em dois subconjuntos de dimensões idênticas, um de treino e outro de teste. De seguida, os papeis invertem-se e o conjunto de treino passa a ser o conjunto de teste e vice-versa. Esta abordagem é chamada de validação cruzada com duas dobras (*two-fold*). A taxa de erro total é a média das taxas de erro das duas rondas [54].

No caso mais geral de validação cruzada com k dobras, o conjunto de dados é segmentado em k subconjuntos de dimensões iguais. Em cada passo do processo, uma das partições é escolhida para teste e as restantes utilizadas para treino. Este procedimento é repetido k vezes, de modo que cada partição seja escolhida exactamente uma só vez para a tarefa de teste. De forma semelhante ao exemplo anterior, a taxa de erro é calculada tomando a média de todas as taxas de erro derivadas de cada um dos k passos.

Um caso particular da validação cruzada é quando $k = m$, *i.e.* o número de dobras é igual ao número de objectos (ver Figura 2.8). Esta abordagem foi utilizada neste trabalho e é designada por validação cruzada *leave-one-out* (LOO-CV[10]). Nesta situação o conjunto de teste tem apenas um exemplo, enquanto o conjunto de treino é composto por $m-1$ exemplos. A principal vantagem deste método é a de utilizar o maior número de instâncias possíveis para treinar o classificador, o que pode aumentar as probabi-

[10] Abreviatura do termo inglês *leave-one-out cross-validation*.

lidades do classificador induzido se tornar mais preciso. Mais ainda, os conjuntos de teste definidos nas diferentes instâncias são mutuamente exclusivos e cobrem completamente o conjunto de dados. A desvantagem mais evidente está ligada ao esforço computacional que é necessário para repetir o procedimento m vezes. Pode-se ainda apontar como desvantagem o facto de o conjunto de teste ser constituído por apenas um exemplo, originando assim uma variância elevada da estimativa do desempenho medido [2, 47].

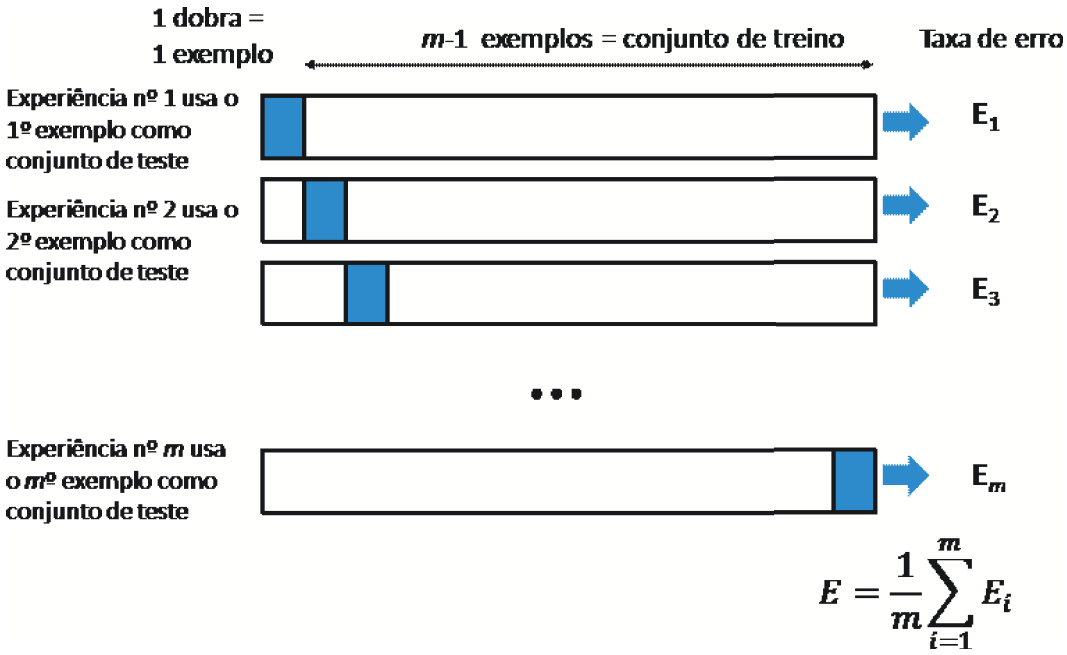


Figura 2.8: Representação esquemática da forma como o método LOO-CV opera num conjunto de dados.

Capítulo 3

Pré-processamento de dados de expressão genética

3.1 Introdução

Para que a análise de dados seja profícua é fundamental realizar um pré-processamento adequado. Em particular, os dados de expressão genética não são excepção. Consoante o tipo de dados existem métodos mais apropriados, ou menos, para o efeito.

Em termos gerais, o pré-processamento de dados reais é realizado em virtude destes serem *(i)* incompletos, *(ii)* inconsistentes ou *(iii)* conterem ruído. Se não existir qualidade nos dados que serão utilizados para futuras análises, estas e posteriores conclusões poderão estar comprometidas.

Genericamente, as principais tarefas de pré-processamento de dados são as seguintes [34]:

- *Limpeza de dados* - inclui a imputação de valores ocultos, alisamento de dados com ruído, identificação e remoção de *outliers* e resolução de inconsistências;
- *Integração dos dados* - combinação de dados provenientes de múltiplas fontes de forma a obter uma junção coerente;
- *Transformação dos dados* - normalização, alisamento, agregação e generalização;

- *Redução da dimensionalidade* - redução da dimensionalidade do espaço de atributos do problema de modo a produzir os mesmos resultados ou similares;

- *Discretização de dados* - conversão de atributos numéricos em categóricos (discretos).

O presente capítulo faz uma introdução aos dois tipos de métodos de pré-processamento usados em experiências de *microarrays* avaliados na presente dissertação. Estes métodos incluem-se na tarefa de transformação dos dados. Primeiramente são abordados os métodos de CB, onde é feita uma descrição do conceito de *background* em experiências de *microarrays*, destacando a importância do processamento de imagem e os consequentes métodos de estimação do valor de *background*. Seguidamente apresenta-se uma perspectiva global de estudos publicados sobre CB e definem-se os métodos de CB considerados. Numa segunda parte, é introduzido o conceito de normalização e é abordado o modelo de regressão *loess* como ferramenta básica na construção de vários procedimentos de NM de dados. Por último, é feita uma descrição de alguns trabalhos recentes relacionados com a NM e são explicados os métodos de NM empregues aos dados no estudo experimental realizado.

3.2 Correcção de *background*

3.2.1 Introdução

No fim da parte laboratorial de uma experiência de *microarrays*, as lâminas são passadas por um *scanner*. O resultado é uma série de imagens digitais com os valores de fluorescência medidos nos vários pontos da lâmina. No caso de *microarrays* de dois canais o resultado é uma imagem digital por canal em tons de cinzento. O *scanner* faz a leitura do *microarray* dividindo-o numa elevada quantidade de *pixels* e guardando a intensidade da fluorescência para cada *pixel* medido na gradação de cores habitual, ou seja, em tons de verde e vermelho.

O primeiro passo na análise de dados de *microarrays* é o processamento de imagem. Na realidade, os dados extraídos das imagens digitais representam os dados primários

recolhidos em cada experiência. Assim, todos os métodos empregues e as análises subsequentes são derivados destas imagens e da sua análise inicial. Deste modo, de entre todos os passos que contribuem para uma experiência de *microarrays* bem sucedida o processamento de imagem é consideravelmente significativo. A maioria dos fabricantes de *scanners* para *microarrays* fornece *software* adequado para lidar com esta etapa da experiência, contudo, é importante perceber como é que os dados são extraídos das imagens digitais.

Em princípio, as intensidades dos *pixels* que não correspondem a pontos do *microarray* deveriam ser zero [1]. No entanto, esta situação raramente acontece face a razões que se prendem, por exemplo, com *(i)* a fluorescência natural do vidro do *microarray*; *(ii)* o revestimento do *microarray* com produtos químicos; *(iii)* a ligação não específica da amostra marcada com composto fluorescente à superfície do *microarray*; entre outros. Apesar desta emissão de fluorescência ser baixa não é negligenciável [1]. Desta forma, é provável que as intensidades medidas nos pontos do *microarray* contenham uma certa quantidade desta fluorescência não específica, designada por fluorescência de *background*.

A fluorescência de *background*, vulgarmente designada apenas por *background*, é medida numa região próxima do ponto do *microarray*, normalmente concêntrica a este mas naturalmente não coincidente com o mesmo. No interior desta região a intensidade é mais forte em virtude de ser nesta secção do ponto que a sonda foi colocada e a hibridação ocorreu. A intensidade dessa região do interior do ponto, denominada por *foreground*, é proporcional à quantidade de hibridação específica.

Assumindo que para *microarrays* de dois canais a intensidade de um ponto é uma combinação aditiva da verdadeira intensidade do ponto e do *background*, o procedimento normal neste tipo de experiências passa pela estimação da intensidade do *background*. Consequentemente, é necessário remover esta parte do sinal que não é devido à hibridação de ADN-complementar à sonda. Esta remoção é conhecida como correcção de *background*.

Seguindo as notações usuais na literatura da especialidade (veja-se, por exemplo, [17]),

denotam-se as intensidades vermelha e verde de *foreground* por R_f e G_f[1], respectivamente, e as intensidades vermelha e verde de *background* por R_b e G_b, respectivamente. Assim, o método base para corrigir os valores dos sinais lidos, R e G, é feito a partir da subtracção das intensidades de *background* às intensidades de *foreground*, isto é, $R = R_f - R_b$ e $G = G_f - G_b$.

3.2.2 Processamento de imagem e métodos de estimação de *background*

Um dos principais objectivos do processamento de imagem de um *microarray* é extrair os valores das intensidades de *foreground* e *background* dos canais vermelho e verde para cada ponto do *microarray*. Após a extração desses valores as intensidades de *background*, R_b e G_b, são usadas para corrigirem as de *foreground*, R_f e G_f. O processamento de imagem é ainda importante para recolher medidas de qualidade para cada ponto do *microarray*. O tópico da qualidade dos pontos de um *microarray* abarca vários passos, no entanto, sai fora do âmbito deste trabalho[2].

O processo de análise da imagem de um *microarray* pode ser subdividida em três tarefas [67]:

- Associação - é o processo de associar coordenadas a cada ponto;
- Segmentação - permite a classificação de *pixels* em *foreground* ou *background*;
- Extracção de intensidades - corresponde ao cálculo das estimativas das intensidades de *background* e de *foreground*, R_b, G_b, R_f e G_f.

A primeira tarefa, a de *associação*, tem o propósito de associar coordenadas ao centro de cada ponto da lâmina de vidro. Aquando da fabricação dos *microarrays*, nomeadamente no passo de impressão dos pontos, raramente existe um perfeito alinhamento dos pontos na lâmina rectangular. Como consequência, não é possível sobrepor uma grelha ao

[1]Para abreviar vermelho (***R**ed*) e verde (***G**reen*) vão ser usadas as iniciais das respectivas palavras em inglês. Quase a totalidade da literatura é em inglês e esta é a notação universalmente adoptada tornando-se assim mais coerente.

[2]Uma apresentação detalhada sobre este último tópico pode ser encontrada em [1].

microarray e ajustá-la até estar absolutamente alinhada. É necessário, então, fazer algumas modificações ao nível do espaçamento entre colunas e entre linhas na grelha sobreposta até estar o alinhamento correcto. Os *software* de processamento de imagem executam esta tarefa de modo quase automatizado. É, no entanto, necessária uma intervenção do investigador para aumentar a fiabilidade dessa associação, no sentido de fornecer informação por exemplo, sobre o número de pontos e o esquema de arranjo do *microarray*. Dos *software* de processamento de imagem utilizados neste tipo de experiências podem-se destacar o Spot [7], o GenePix [25] e o QuantArray [49].

Após a tarefa de associação, é preciso segmentar os *pixels* em duas categorias: os que se encontram de facto no interior do ponto, *foreground*, e os que se encontram na região circundante, *background*. Existem diferentes métodos para realizar esta segmentação que se encontram detalhadamente explicados em [67]. Estes podem-se agrupar em quatro grupos, consoante a geometria dos pontos que produzem:

- segmentação de círculos fixos;
- segmentação de círculos ajustados;
- segmentação de formas ajustadas;
- segmentação pelo método do histograma.

Uma vez identificados os *pixels* de *foreground*, as intensidades de *foreground* para um dado ponto são geralmente estimadas como sendo a média da intensidade de todos os *pixels* de *foreground*. Esta intensidade deve, por sua vez, ser directamente proporcional ao número de moléculas de ADN-complementar hibridadas com as sondas em cada ponto do *microarray* [60].

As intensidades de *background* são habitualmente estimadas usando a mediana das intensidades [60]. Para a estimar as intensidades de *background*, há diversas opções. Destacam-se as seguintes [67]:

- *Background* local: as intensidades de *background* são estimadas tendo em conta pequenas regiões na zona circundante de cada ponto. As diferentes regiões, conforme ilustra a Figura 3.1, são obtidas considerando:

  - todos os *pixels* que estão entre o círculo que define o ponto e uma caixa limitativa que está centrada no ponto. A estimativa é assim feita pelo cálculo da mediana dos valores desses *pixels*;
  - a área entre duas circunferências concêntricas centradas no ponto;
  - quatro áreas com forma de losango, denominadas por vales do *microarray* e que se localizam nos quatro cantos de cada ponto. Neste caso o *background* local é calculado como sendo a mediana dos valores dos quatro vales.

- Abertura morfológica: este método está implementado no *software* Spot. Resumidamente, este remove todos os pontos do *microarray* e gera uma imagem que determina a estimativa do *background* para cada ponto da lâmina. Essa estimativa é obtida utilizando um filtro não linear chamado de abertura morfológica.

- *Background* constante: este é um método global, no sentido em que é subtraído um *background* constante a todos os pontos do *microarray*. O valor constante é por vezes definido como o terceiro percentil (3%) dos valores de *foreground* de todos os pontos.

- Sem estimação: é possível não estimar qualquer valor para as intensidades de *background*, o que significa usar, para as análises subsequentes, os valores de *foreground* como os valores verdadeiros, *i.e.* $R = R_f$ e $G = G_f$.

Segundo os trabalhos de investigação na área de processamento de imagem, [67] e [69], a escolha do método de estimação de intensidades de *background* tem um maior impacto na determinação dos valores de expressão genética do que o passo de segmentação.

3.2.3 Literatura relacionada

Nesta secção, pretende-se criar uma perspectiva global sobre a grande quantidade de trabalhos que têm sido realizados no contexto do estudo de metodologias de correcção de *background*, CB.

Segundo [67] e [69], os métodos de estimação de *background*, baseados em médias ou medianas das intensidades nas regiões vizinhas dos pontos do *microarray*, tendem a

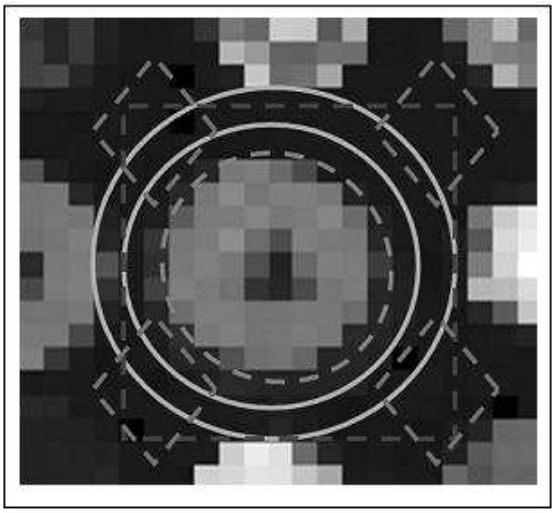


Figura 3.1: Imagem ilustrativa de diferentes estimativas de *background* local. A região no interior da circunferência a ponteado representa um ponto de um *microarray*. As outras linhas representam regiões usadas para os diferentes métodos de estimação de *background* local. Quadrado a ponteado: usada pelo primeiro método de *background* local. Circunferências de linha sólida: usadas pelo segundo método de *background* local. Losangos a ponteado: usados pelo terceiro método de *background* local (figura extraída de [67]).

produzir estimativas que serão afectadas por ruído. Deste modo, ao fazer a correcção de *background*, há possibilidade de aumentar a variabilidade dos logaritmos da razão da expressão genética, $\log_2 R/G$ [3]. No entanto, aqueles autores também não aconselham a não correcção dos valores de intensidade com respeito ao *background*. Por isso, uma vez que a correcção de *background* é realizada pela subtracção dos valores de *background* aos de *foreground* é recomendado, em alternativa, um ajuste de *background* intermédio. Espera-se que tal ajuste evite uma certa variabilidade provenientes de métodos de estimação baseados em medidas de localização e, por outro lado, que torne as estimativas mais precisas que os valores de intensidade brutos.

Em [67] chama-se ainda a atenção que, no caso da estimativa segundo o método da abertura morfológica não estar disponível no *software* de apoio ao processamento de imagem, os valores de intensidade lidos pelo *scanner* não devem ser corrigidos pela subtracção do *background*. Também num outro estudo é recomendada a não subtracção dos valores de *background* pelas razões do aumento de variabilidade dos dados de

[3] Na Secção 3.3 será detalhadamente explicada esta expressão.

expressão genética [50].

O estudo de simulação de expressão diferencial [56] teve como objectivo mostrar a relação do viés, da variância e do erro quadrático médio com a aplicação, ou não, de subtracção de *background*. A decisão de aplicar a subtracção de *background v.s.* não aplicar subtracção de *background* resultou de um compromisso entre o viés e a variância que foi atingido através dos dados simulados que preveniram a introdução da variabilidade técnica. Estes autores concluem que a correlação dos valores de *foreground* e *background* é uma consideração importante a ter em conta antes da subtracção da intensidade de *background*. Outra recomendação ainda indicada nesse estudo é utilização de gráficos exploratórios úteis para ajudar na decisão de subtrair as estimativas de *background* local. Exemplos desses gráficos são as imagens da fluorescência de *background* e os gráficos de dispersão entre as intensidades de *foreground* e *background*.

Qualquer método de estimação das intensidades de *background* pode originar estimativas de valor superior às de *foreground*, ou seja, $R_f < R_b$ ou $G_f < G_b$. Como consequência, os valores finais das intensidades verde e vermelha tornam-se negativos, assim, pode ocorrer que $R/G < 0$. Este facto tem consequências problemáticas quando é necessário calcular logaritmos dos valores R/G, tornando a subtracção de *background* um método de CB limitativo.

Vários estudos têm investigado a melhor forma de lidar com as intensidades de *background* no panorama geral das intensidades medidas pelos *scanners*. Concretamente, questiona-se se é mais vantajoso, no decorrer das análises e posteriores conclusões, *(i)* proceder a um pré-processamento dos valores medidos tendo em conta o efeito de *background* ou *(ii)* trabalhar com os valores em bruto. Existem investigadores que depois das recomendações dos estudos atrás mencionados ([50, 67, 69]) optaram, nos seus próprios trabalhos, pela opção *(ii)*, ou seja, por não fazerem correcção de *background*, como por exemplo [76]. Todavia, para os investigadores para os quais a opção *(ii)* não é suficiente e para os quais a subtracção de *background* é limitativa, iniciaram o desenvolvimento de outros métodos para realizar a correcção de *background*.

O método Kooperberg [38], é um exemplo de um método de CB criado para fazer face aos problemas mencionados no parágrafo anterior. Este método sugere um modelo empírico Bayesiano construído com base em convoluções de distribuições normais para

corrigir as intensidades lidas através dos *scanners* relativamente ao *background*. Um método igualmente baseado em convoluções, no entanto, entre uma distribuição normal e uma exponencial, método *normexp*, foi desenvolvido e já adaptado por diversos autores [4, 33, 52]. Um outro exemplo é o método Edwards [18] que propõe evitar a ocorrência de valores negativos de intensidade através de uma função monótona suave. Estes dois últimos métodos serão abordados no presente estudo.

Propostos diferentes métodos de CB, naturalmente surgem trabalhos de comparação dos mesmos. Em [4], usando dados de *microarrays* de um canal, uma comparação de diferentes métodos de CB é feita com base em medidas de precisão e exactidão e na capacidade de detectar conjuntos de sondas diferencialmente expressas.

Em [52], há um estudo comparativo de métodos combinados de estimação e correcção de *background* em dados de *microarrays* de dois canais. A comparação das diferentes combinações é realizada em função de quão bem cada uma executa a escolha de genes diferencialmente expressos através da análise de curvas ROC[4]. Os genes diferencialmente expressos são seleccionados através de uma ordenação dos mesmos, usando uma estatística que pode ser encontrada em [62]. A avaliação dos métodos é feita com base na análise de variância dos logaritmos dos valores da razão da expressão genética e do viés. É também realizado um estudo dos valores omissos aos dados para cada conjunto de métodos de estimação e correcção de *background*.

Um trabalho do mesmo autor de [52], continua a explorar a comparação entre diferentes métodos de CB [53]. Contudo, a medida de desempenho tem uma ligeira adaptação ao nível dos algoritmos para descoberta de genes diferencialmente expressos. Os algoritmos SAM[5] [65] e limma eBayes [62], bastante estudados na literatura, foram avaliados com base na razão de falsas descobertas. Da mesma forma é realizada uma análise ao nível da variância e viés dos logaritmos da razão da expressão genética.

[4] ***R**eceiver **O**perator **C**haracteristic*
[5] ***S**ignificance **A**nalysis of **M**icroarrays*

3.2.4 Métodos de correcção de *background*

Nesta secção descrevem-se os seis métodos de CB aplicados aos dados em estudo no Capítulo 4, encontrando-se na Tabela 3.1 os comandos existentes no pacote *limma* do *software* R para esses métodos.

Método	Função com especificação do parâmetro	Abreviatura
None	**backgroundCorrect**(RGlist,method="none")	NB
Subtraction	**backgroundCorrect**(RGlist,method="sub")	sub
Minimum	**backgroundCorrect**(RGlist,method="min")	min
Half	**backgroundCorrect**(RGlist,method="half")	half
Edwards	**backgroundCorrect**(RGlist,method="edwards")	edw
Normexp	**backgroundCorrect**(RGlist,method="normexp")	nexp

Tabela 3.1: Comandos usados no *software* R/*Bioconductor* através do pacote *limma* para os seis métodos de CB aplicados às bases de dados analisadas no Capítulo 4.

None (NB):

Nenhum método de CB é aplicado, o que significa que $R = R_f$ e $G = G_f$.

Subtraction (sub):

A correcção é efectuada através da subtracção dos valores de *background* pelos valores de *foreground*, tais que $R = R_f - R_b$ e $G = G_f - G_b$.

Half (half):

Neste método toda a intensidade que é menor do que 0.5, depois da subtracção das intensidades de *background* às de *foreground*, é ajustado para o valor 0.5, caso contrário mantém-se a subtracção do método anterior. Concretamente,

$$G = \begin{cases} G_f - G_b & , \text{ se } G_f - G_b \geq 0.5 \\ 0.5 & , \text{ caso contrário} \end{cases}$$

e

$$R = \begin{cases} R_f - R_b & , \text{ se } R_f - R_b \geq 0.5 \\ 0.5 & , \text{ caso contrário} \end{cases}$$

Minimum (min):

Toda a intensidade que seja negativa, depois da subtracção das intensidades de *background* às de *foreground*, é ajustada para a menor intensidade corrigida não negativa do conjunto de sondas, caso contrário mantém-se a subtracção do método sub, *i.e.*,

$$G = \begin{cases} G_f - G_b & , \text{ se } G_f - G_b \geq 0 \\ \min_{1 \leq i \leq N}\{G_{f_i} - G_{b_i} : G_{f_i} - G_{b_i} \geq 0\} & , \text{ caso contrário} \end{cases}$$

e

$$R = \begin{cases} R_f - R_b & , \text{ se } R_f - R_b \geq 0 \\ \min_{1 \leq i \leq N}\{R_{f_i} - R_{b_i} : R_{f_i} - R_{b_i} \geq 0\} & , \text{ caso contrário} \end{cases}$$

onde N é o número de sondas no *microarray*.

Edwards (edw):

Este método foi estabelecido no trabalho [18] onde se define a correcção das intensidades do seguinte modo:

$$G = \begin{cases} G_f - G_b & , \text{ se } G_f - G_b \geq \delta \\ \delta e^{1-(G_b+\delta)/G_f} & , \text{ outros casos} \end{cases}$$

e

$$R = \begin{cases} R_f - R_b & , \text{ se } R_f - R_b \geq \delta \\ \delta e^{1-(R_b+\delta)/R_f} & , \text{ outros casos} \end{cases}$$

Neste método [18], a subtracção de *background* é realizada de forma natural quando a diferença entre as intensidades de *foreground* e *background* é superior a um determinado valor δ. No entanto, quando a diferença é inferior ao limite estabelecido, a subtracção é substituída por uma função monótona suave. O valor de δ depende dos dados concretos e pode ser consultado na função *backgroundCorrect* do *software R*.

Normexp (nexp):

Este método é baseado na convolução das distribuições normal e exponencial [4, 33, 52]. A intensidade corrigida passa a ser o valor esperado da intensidade verdadeira dado que

é conhecida a intensidade de *foreground* observada. Denotando as variáveis aleatórias:

$$\begin{aligned} O &= \text{Sinal de } foreground \text{ observado,} \\ V &= \text{Sinal verdadeiro,} \\ BG &= \text{Sinal de } background, \end{aligned}$$

este modelo assume que $V \sim Exp(\alpha)$, $BG \sim \mathcal{N}(\mu, \sigma)$ e $O = V + BG$, com V e BG independentes. Mais ainda, assume que $BG \geq 0$ para evitar produzir valores negativos [4]. Nestas condições, BG é normalmente distribuída truncada em zero, ou seja,

$$f_{BG}(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{1}{2\sigma^2}(x-\mu)^2 \Big) \frac{1}{1 - \Phi(0; \mu, \sigma)}$$

onde Φ é a função de distribuição da normal de parâmetros (μ, σ).

Sob este modelo, os valores de *background* corrigidos serão dados por, $E(V|O)$.

A suposição para a distribuição exponencial foi baseada nas densidades das intensidades de *foreground* observadas num certo número de experiências de *microarrays* [4, 53]. Estas densidades empíricas mostraram curvas enviesadas para a direita com longas caudas semelhantes às da densidade de uma distribuição exponencial. Adicionalmente, existe a vantagem de serem conhecidas propriedades relativas à convolução de uma distribuição normal com uma exponencial. Este facto conduz a expressões bem definidas do valor esperado do sinal verdadeiro dado que são conhecidos os valores das intensidades de *foreground* observadas [11].

A distribuição conjunta de (V, BG) é:

$$\begin{aligned} f_{(V,BG)}(x, y; \alpha, \mu, \sigma) &= f_V(x; \alpha) f_{BG}(y; \mu, \sigma) \\ &= \frac{1}{\alpha} \exp\Big(-\frac{x}{\alpha} \Big) \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{1}{2\sigma^2}(y-\mu)^2 \Big) \frac{1}{1 - \Phi(0; \mu, \sigma)}, \ x, y > 0 \end{aligned}$$

A distribuição conjunta de (V, O) é consequentemente:

$$\begin{aligned} f_{(V,O)}(x, z; \alpha, \mu, \sigma) &= f_{(V,BG)}(x, z - x; \alpha, \mu, \sigma) \\ &= \frac{1}{\alpha} \exp\Big(-\frac{x}{\alpha} \Big) \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{1}{2\sigma^2}(z-x-\mu)^2 \Big) \frac{1}{1 - \Phi(0; \mu, \sigma)} \\ &= \frac{1}{\alpha\sqrt{2\pi\sigma^2}(1 - \Phi(0; \mu, \sigma))} \exp\Big(-\frac{x}{\alpha} - \frac{1}{2\sigma^2}(z-x-\mu)^2 \Big) \end{aligned}$$

Os cálculos auxiliares que se seguem são precisos para a simplificação da distribuição conjunta (V, O):

$$\begin{aligned}
-\frac{x}{\alpha} - \frac{1}{2\sigma^2}(z-x-\mu)^2 &= -\frac{x}{\alpha} - \frac{1}{2\sigma^2}(x-(z-\mu))^2 \\
&= -\frac{1}{2\sigma^2}\Big[x^2 + (z-\mu)^2 - 2x\big(z-\mu-\frac{\sigma^2}{\alpha}\big)\Big] \\
&= -\frac{1}{2\sigma^2}\Big[x^2 + \Big(\big(z-\mu-\frac{\sigma^2}{\alpha}\big) + \frac{\sigma^2}{\alpha}\Big)^2 - 2x\Big(z-\mu-\frac{\sigma^2}{\alpha}\Big)\Big] \\
&= -\frac{1}{2\sigma^2}\Big[x - \Big(z-\mu-\frac{\sigma^2}{\alpha}\Big)\Big]^2 - \frac{(z-\mu)}{\alpha} + \frac{\sigma^2}{2\alpha^2} \\
&= -\frac{1}{2\sigma^2}(x-\mu_z)^2 - \frac{(z-\mu)}{\alpha} + \frac{\sigma^2}{2\alpha^2}
\end{aligned}$$

onde $\mu_z = z - \mu - \frac{\sigma^2}{\alpha}$

Depois desta simplificação, a distribuição conjunta (V, O) vem dada por:

$$f_{(V,O)}(x, z; \alpha, \mu, \sigma) = \frac{1}{\alpha\sqrt{2\pi\sigma^2}(1-\Phi(0;\mu,\sigma))} \exp\Big(-\frac{(z-\mu)}{\alpha} + \frac{\sigma^2}{2\alpha^2}\Big) \exp\Big(-\frac{1}{2\sigma^2}(x-\mu_z)^2\Big)$$

A distribuição marginal de O resulta de integrar a distribuição (V, O) conjunta em relação a x no seu suporte:

$$\begin{aligned}
f_O(z) &= \int_0^\infty f_{(V,O)}(x, z)dx \\
&= \frac{1}{\alpha(1-\Phi(0;\mu,\sigma))} \exp\Big(-\frac{(z-\mu)}{\alpha} + \frac{\sigma^2}{2\alpha^2}\Big) \int_0^\infty \frac{1}{\sqrt{2\pi\sigma^2}} \exp\Big(-\frac{1}{2\sigma^2}(x-\mu_z)^2\Big)dx \\
&= \frac{\exp\Big(-\frac{(z-\mu)}{\alpha} + \frac{\sigma^2}{2\alpha^2}\Big)}{\alpha(1-\Phi(0;\mu,\sigma))}(1-\Phi(0;\mu_z,\sigma))
\end{aligned}$$

O propósito destes passos intermédios é determinar a distribuição condicional de V dado O, porque como foi dito anteriormente, os valores corrigidos pelo *background* são os valores esperados de V dado O. A distribuição condicional V dado O é então:

$$f_{V|O}(x|z; \alpha, \mu, \sigma) = \frac{f_{(V,O)}(x, z; \alpha, \mu, \sigma)}{f_O(z; \alpha, \mu, \sigma)} = \frac{\frac{1}{\sqrt{2\pi\sigma^2}}\exp(-\frac{1}{2\sigma^2}(x-\mu_z)^2)}{1-\Phi(0;\mu_z,\sigma)}$$

para $x > 0$.

Determine-se agora o valor esperado de V dado O:

$$E(V|O) = \int_0^\infty x \frac{\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(x-\mu_z)^2\right)}{1-\Phi(0;\mu_z,\sigma)} dx$$

Repare-se que dentro do integral tem-se a expressão da função densidade de uma distribuição normal truncada em zero com parâmetros (μ_z,σ). Assim, tem-se,

$$E(V|O) = \mu_z + \sigma^2 \frac{\frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{\mu_z^2}{2\sigma^2}\right)}{1-\Phi(0;\mu_z,\sigma)}$$

Os parâmetros α, μ e σ podem ser estimados pelo método de máxima verosimilhança usando a função densidade da variável O,

$$\log f_O(z;\alpha,\mu,\sigma) = -\log\alpha - \log(1-\Phi(0;\mu,\sigma)) - \frac{z-\mu}{\alpha} + \frac{\sigma^2}{2\alpha^2} + \log(1-\Phi(0;\mu_z,\sigma))$$

3.3 Normalização

3.3.1 Introdução

O segundo tipo de métodos de pré-processamento usualmente aplicado a dados de *microarrays* são os métodos de normalização (NM). Neste capítulo serão abordados vários tópicos relacionados com o conceito de normalização.

3.3.1.1 Notação

Na Secção 3.2 foi introduzida a notação para as intensidades de *background* e *foreground* verdes e vermelhas e foram frequentemente mencionados os "logaritmos da razão da expressão genética"sem nunca serem definidos. Faz sentido proceder à sua formalização nesta secção uma vez que toda ela se baseia nesses valores.

Considerem-se as intensidades vermelha e verde finais, quer haja CB quer não, R e G respectivamente. A intensidade da razão da expressão genética, é dada pelo quociente R/G. Para cada ponto do *microarray* a intensidade da razão da expressão genética refere-se a um ponto do mesmo, assim como obviamente as intensidades R_f, G_f, R_b, G_b.

Na verdade, a razão R/G é raramente utilizada sendo prática usual trabalhar com o logaritmo de base 2 da razão das intensidades[6]. Assim, em vez de R/G passa a considerar-se seguinte fórmula

$$M = \log_2(R/G) \tag{3.1}$$

Para cada ponto passa-se a ter o logaritmo da razão da expressão genética, M. Outra expressão também muito utilizada é

$$A = \log_2\sqrt{RG} \tag{3.2}$$

e é uma medida da intensidade total de um ponto do *microarray*. M surge como mnemónica para *menos*, do inglês *minus*, pois $M = \log_2 R - \log_2 G$, enquanto A é uma mnemónica para *soma*, do inglês *add*, uma vez que $A = \frac{1}{2}\log_2(RG) = \frac{\log_2 R + \log_2 G}{2}$.

O facto de serem usados os logaritmos das intensidades, $\log_2 R$ e $\log_2 G$, em vez dos valores das intensidades, R e G, justifica-se por várias razões [17, 60]. Entre elas estão, o facto da variação dos logaritmos das intensidades e dos logaritmos das razões das intensidades serem menos dependentes da magnitude absoluta dos valores medidos. Note-se que as intensidades medidas pelos *scanners* pertencem a um intervalo entre 0 e $2^{16} - 1 = 65535$, pois as imagens vêm num formato de 16-bit. Assim, em virtude de uma vasta maioria das intensidades medidas serem menores que 1000, se os dados não forem de alguma forma compactados, tornam-se imperceptíveis as intensidades mais baixas em gráficos com escalas de magnitude elevada, Figuras 3.2(a) e 3.2(b). Outra explicação está ligada ao facto da normalização ser aditiva para os logaritmos das intensidades o que se torna importante no sentido da simplificação de cálculos, ou seja, os logaritmos convertem as razões R/G em diferenças $M = \log_2 R - \log_2 G$.

Uma análise exploratória dos dados, mais concretamente através da representação gráfica dos mesmos, pode auxiliar a perceber se o ensaio foi bem sucedido e, para além disso, identificar problemas específicos, que podem ser resolvidos com uma escolha apropriada de ferramentas para lidar com tais situações. Existem várias maneiras de representar os valores das intensidades através de gráficos, sendo uma das mais comuns, o gráfico de dispersão entre os valores dos logaritmos da intensidade vermelha,

[6] Os logaritmos de base 2 são usados em vez de logaritmos de base 10 ou de base e pelo facto das intensidades assumirem valores inteiros entre 0 e $2^{16} - 1$ [60].

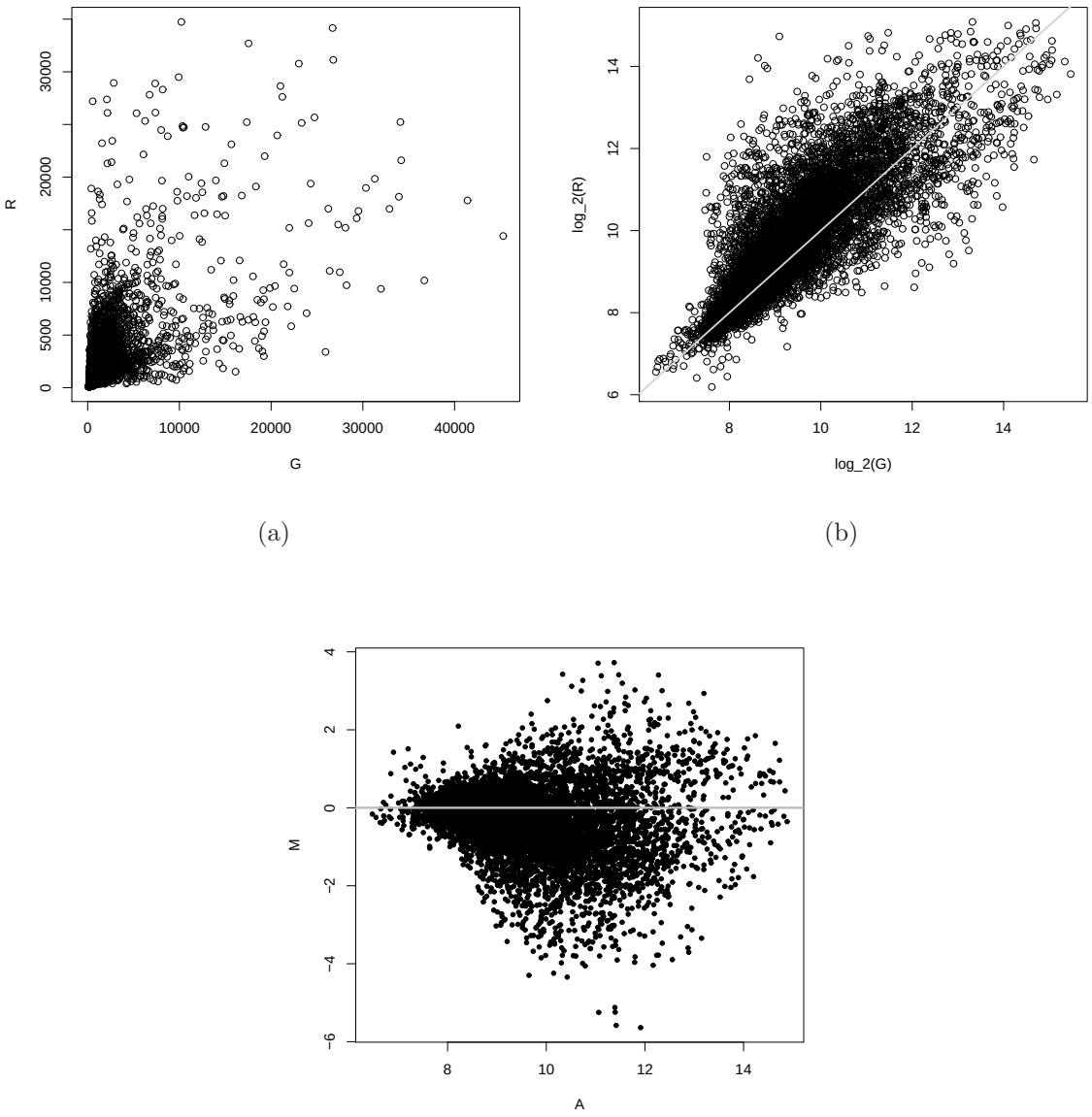
$\log_2 R$, *versus* os logaritmos da intensidade verde, $\log_2 G$, veja-se a Figura 3.2(b). Apesar desta opção ser bastante directa, de forma geral há uma grande correlação entre as intensidades dos dois canais [60], o que torna complicado distinguir características importantes intrínsecas aos dados. Repare-se que o interesse reside nos desvios dos pontos em relação à linha diagonal $\log_2 R = \log_2 G$, onde a expressão em cada canal é a mesma. Desta forma torna-se vantajoso fazer uma rotação de 45°e redimensionar os eixos no novo gráfico. O novo gráfico é designado por gráfico-MA [17], pois em vez de representar $\log_2 R$ *vs.* $\log_2 G$, representa M *vs.* A, veja-se a Figura 3.2(c). Este gráfico torna mais perceptível a procura de relações não lineares entre os logaritmos das intensidades. Mais ainda, tornam-se mais imediatas essas diferenças em relação à recta que traduz a igualdade das intensidade nos dois canais, $M = 0$.

3.3.1.2 Fontes de viés

As experiências de *microarrays*, como qualquer outro processo laboratorial, estão sujeitas a erros. Sendo estas complexas, o processo experimental frequentemente introduz efeitos sistemáticos não desejados nas medidas das intensidades. Em [1] é salientado que estes efeitos podem ser de tal forma substanciais que diluem os efeitos que os investigadores procuram.

O exemplo dado como clássico para explicitar este tipo de acontecimentos são as hibridações *self-self* [20]. Estas são ensaios onde a mesma amostra é usada para ser comparada consigo própria. Neste tipo de experiência duas amostras idênticas de ADN-complementar são marcadas, cada uma individualmente, com os fluoróforos Cy3 e Cy5 e são hibridadas na mesma lâmina. Aqui é esperado que o logaritmo da razão das intensidades medidas, M, seja zero para cada gene. Isto é, que o quociente R/G seja 1, uma vez que não deveria haver nenhuma diferença entre a representação dos genes no ARN inicial. É expectável não existir nenhuma expressão diferencial e, consequentemente, os valores-M serem zero, ou seja, não haver nenhuma alteração do nível de expressão dos genes quando a condição de interesse é induzida.

Na Figura 3.3 está representado o gráfico-MA da hibridação *self-self* estudada em [58]. Este gráfico mostra claramente uma nuvem de pontos com uma curvatura ligeiramente ascendente nas intensidades mais baixas e ainda uma dispersão razoavelmente elevada



(a) (b)

(c)

Figura 3.2: Gráficos resultantes dos dados do *microarray* #6039 da base de dados Lymphoma a ser usada no Capítulo 4. (a) Gráfico R *vs.* G; (b) Gráfico $\log_2 R$ *vs.* $\log_2 G$; (c) Gráfico-MA. As linhas $\log_2 R = \log_2 G$ e $M = 0$ estão representadas como referência.

na distribuição dos valores-M nas intensidades baixas.

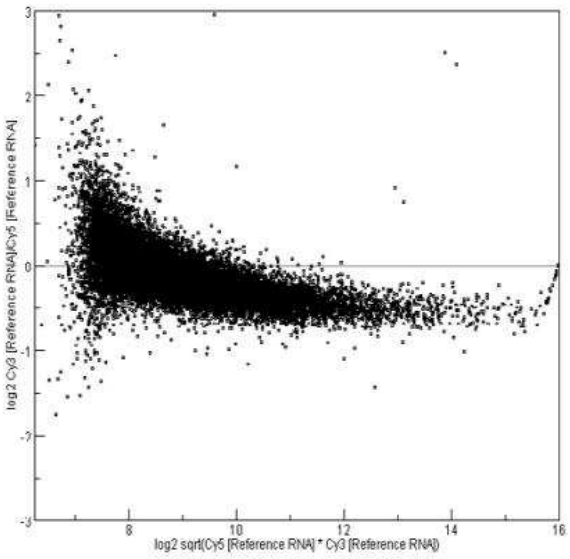


Figura 3.3: Gráfico-MA de uma hibridação *self-self* (figura extraída de [58]).

É conhecido ainda que o "viés introduzido pelo fluoróforo"[7] está presente em quase todas as experiências de *microarray* de multicanais [1]. Usualmente as intensidades representadas com vermelho (Cy5), tendem a ser superiores às representadas por verde (Cy3), mas a grandeza das diferenças geralmente depende da intensidade total medida. A desigualdade entre as intensidades dos dois canais explica-se com base nas propriedades físico-químicas dos compostos fluorescentes, nas eficiências da marcação das amostras com os fluoróforos, entre outras [1]. O viés introduzido pelo fluoróforo é função das intensidades medidas (ver Figura 3.2(c)) e, geralmente, também varia com a posição espacial da lâmina. Estes efeitos podem resultar de diversos factores, entre eles estão, erros ocorridos durante a extracção e isolação do ARN, variações na introdução do composto fluorescente, erros ocorridos durante as medições pelos aparelhos ópticos, inserção não horizontal do *microarray* no *scanner*, lavagem transversal do *microarray* de forma desigual, entre outros [30].

As variações sistemáticas incidem sobre diferentes *microarrays* de modo distinto. Deste modo, para se fazerem comparações válidas sobre os *microarrays* de uma mesma experiência é necessária a remoção dos efeitos dessas variações sistemáticas. Consequentemente, essa eliminação possibilita que os dados originários dos diferentes *microarrays*

[7]Tradução adoptada para o termo inglês *dye bias*.

possam ser comparados sobre uma mesma escala.

A definição de normalização é, concretamente, qualquer tranformação nos dados que os ajusta para as fontes sistemáticas de variação. Em [1] é referido que a normalização pode ser tida como uma espécie de processo de calibração que melhora a comparabilidade entre *microarrays* de forma a serem tratados como iguais.

É de notar que todas as estratégias de normalização são baseadas em suposições subjacentes aos dados e ao processo experimental, pelo que a técnica de normalização precisa de ser usada, de forma apropriada, consoante cada experiência em concreto.

3.3.2 Regressão *loess*

Todos os métodos de NM aplicados aos dados em estudo no Capítulo 4 baseiam-se no modelo de regressão polinomial localmente ponderada, abreviadamente conhecida na literatura inglesa por *loess*. Deste modo, faz-se nesta secção uma apresentação dos tópicos mais importantes da teoria de base desse modelo.

O modelo de regressão *loess* foi originalmente proposto por Cleveland (1979) [13] e mais tarde desenvolvido por Cleveland e Devlin (1988) [14]. Apesar de ser um modelo com fundamentos teóricos bastante conhecidos, existe alguma confusão à volta do acrónimo universalmente utilizado. Se, em alguns trabalhos de aplicação deste método, é possível encontrar o acrónimo *loess*, noutros encontra-se *lowess*. A designação *lowess*, tido por muitos como *LOcally WEighted polynomial regreSSion*, é no entanto também referido como *locally weighted regression scatterplot smothing*. Nesta dissertação será adoptada a versão *loess*, pelo facto de nos artigos originais de Cleveland ser usado este acrónimo. Contudo, tanto de *LOcal regrESSion* como *LOcally weighted regrESSion* são os nomes por extenso do acrónimo *loess* usado nos trabalhos originais [14, 15]. Como consequência desta confusão, por exemplo, no *software R* o comando primordial para executar esta regressão era `lowess()` e mais tarde foi substituído por `loess()`. O que importa destacar é que independentemente do acrónimo a ser usado a teoria que está por detrás é a mesma.

O modelo de regressão *loess* ajusta modelos de regressão, em geral linear ou quadrático, ponderado a subconjuntos locais do conjunto total de dados. O objectivo é construir

uma função global que descreva, ponto por ponto, a relação determinística entre as variáveis independentes e dependente. É de notar que, neste modelo, não há a construção explicita da função global que ajusta um modelo aos dados mas apenas da que ajusta segmentos do conjunto de dados. Outro facto importante é que este é um método não paramétrico, ou seja, não há nenhuma suposição quanto à distribuição de probabilidade do mecanismo subjacente ao fenómeno observável.

Para cada ponto do conjunto de dados um polinómio de baixo grau é ajustado através dos valores das variáveis explicativas próximas do ponto cuja resposta se pretende estimar. Convém notar que o ajuste polinomial é diferente para cada ponto. O polinómio é determinado usando o método dos mínimos quadrados ponderados. Desta forma, pelas propriedades da função peso, será dado mais peso aos pontos próximos do ponto cuja resposta se pretende estimar e menos aos pontos mais distantes. O ajuste *loess* fica completo após os valores da função de regressão terem sido calculados para todos os pontos do conjunto de dados.

Assuma-se, no caso bidimensional, que

$$y_i = f(x_i) + \epsilon_i$$

onde y_i é a variável resposta, f a função de regressão, x_i é a variável independente e ϵ_i é o erro aleatório associado à variável resposta, com $i = 1, ..., n$, onde n representa o número total de pontos do conjunto de dados.

A curva de regressão *loess* é calculada tendo em conta dois parâmetros: o parâmetro alisador[8], α, que varia entre 0 e 1, e o grau do polinómio local, d, normalmente 1 ou 2. Seja $r = \alpha n$ arredondado ao inteiro mais próximo. A função a minimizar no método dos mínimos quadrados ponderados é a seguinte

$$\sum_{i=1}^{n} w_i(x_k)(y_i - f(x_i))^2$$

Assuma-se que se pretende fazer o ajuste no ponto x_k e $w_i(x_k)$ são os pesos para os pontos na vizinhança de x_k.
As seguintes distâncias são calculadas

$$d_i = |x_k - x_i|, i = 1, ..., n, \ i \neq k$$

[8]Tradução do termo inglês *smoothing parameter*, também designado por *span*.

Para cada x_k seja h_k a distância de x_k ao r-ésimo vizinho mais próximo de x_k. Isto é, h_k é o r-ésimo menor número de entre todas as distâncias d_i para $i = 1, ..., n$.

A função de pesos geralmente mais utilizada é a função tricúbica definida por

$$W(u) = \begin{cases} (1 - |u|^3)^3 & , |u| < 1 \\ 0 & , |u| \geq 1 \end{cases} \tag{3.3}$$

O objectivo da função de pesos é tornar a regressão *loess* num ajuste polinomial que tem em conta os vizinhos do ponto (x_k, y_k).

Os pesos para cada ponto são dados por,

$$w_i(x_k) = W\left(\frac{d_i}{h_k}\right)$$

Assim, os pesos dos pontos mais afastados de (x_k, y_k) serão 0. A função W é centrada em x_k e é feito um dimensionamento tal que o primeiro ponto para o qual W se torna nulo seja o r-ésimo vizinho mais próximo de x_k.

Uma vez determinados os pesos $w_i(x_k)$, a função de regressão *loess*, $f(x_k)$, é determinada para o ponto (x_k, y_k) com base no método dos mínimos quadrados ponderados. Todo este procedimento é repetido para cada ponto $(x_k, y_k), k = 1, ..., n$.

Grau do polinómio

Os polinómios locais ajustados a cada subconjunto dos dados são, na maior parte das situações, polinómios de grau 1 ou grau 2, isto é, lineares ou quadráticos. A escolha do grau do polinómio como sendo 1 aparece como um bom balanço entre o esforço computacional e a necessidade da flexibilidade relacional (entre a variável dependente e a independente) para reproduzir o modelo intrínseco aos dados [13]. O caso mais simples ao nível computacional acontece quando o grau do polinómio é 0, todavia, em situações práticas uma suposição de linearidade local será mais abrangente do que uma suposição de estacionaridade local, pois a finalidade é representar variáveis que estejam relacionadas uma com a outra. Na circunstância do grau do polinómio ser 2, e apesar de ser uma situação muitas vezes escolhida, pode em termos computacionais ter tendência a sobrepor-se à necessidade da flexibilidade relacional [13]. Polinómios de

grau superior a 2 em teoria poderiam funcionar mas, por outro lado, iriam representar modelos que não estão no espírito da regressão *loess*. Isto é, qualquer função pode ser aproximada numa pequena vizinhança por um polinómio de baixo grau, ou seja, usam-se funções simples para aproximar conjuntos de dados globalmente complexos [77, 78].

Função de pesos

Em [13] são descritas as propriedades que a função de pesos deve satisfazer. Seja W uma função de pesos com as seguintes propriedades:

1. $W(x) > 0$, para $|x| < 1$;
2. $W(-x) = W(x)$;
3. $W(x)$ é uma função não crescente para $x \geq 0$;
4. $W(x) = 0$, para $|x| \geq 1$.

A primeira propriedade é necessária uma vez que pesos negativos não fazem sentido; a segunda é exigida pelo facto dos pontos à direita de x serem tratados de forma igual aos da esquerda; a terceira é requerida em virtude de não fazer sentido que um ponto mais afastado de x tenha maior peso do que um ponto mais próximo; a última é necessária por razões ao nível computacional [13]. Mais ainda, é desejável que $W(x)$ decresça suavemente para 0 à medida que x varia de 0 para 1.

Uma função de pesos frequentemente usada é a função tricúbica, dada pela Função 3.3. Qualquer outra função de pesos que satisfaça as propriedades descritas pode naturalmente ser usada.

Parâmetro alisador

O parâmetro alisador, α, varia entre 0 e 1 e representa a proporção de pontos do conjunto de dados usados para ajustar cada polinómio local. O objectivo na selecção de α é escolher o maior valor possível que minimize a variabilidade nos pontos estimados sem alterar o modelo intrínseco aos dados [13].

Quando α é um número reduzido, menos pontos ficam envolvidos no ajustamento de um determinado ponto, assim, mais informação local fica reflectida na curva estimada. Por outro lado, se o parâmetro alisador é um número mais elevado isso reflecte-se no tipo de informação que passa para a curva ajustada, ou seja, informação mais global é passada para a curva tornando-a mais suave. Quando este parâmetro é mais elevado acontece haver uma diminuição do efeito de *outliers*, porém, se o valor é demasiado grande alguma informação local importante pode ser perdida.

Na maioria das situações práticas a escolha de α varia entre 0.2 e 0.8, contudo, em situações onde não há nenhum indício do valor de α necessário, $\alpha = 0.5$ é um bom ponto de partida [13]. A prática comum para dados de *microarrays* é tomar $\alpha = 0.4$ [69].

Vantagens e desvantagens do modelo de regressão *loess*

Este modelo não necessita de uma especificação explícita da função global que define o modelo ajustado aos dados. Como alternativa é apenas necessário fornecer o parâmetro alisador e o grau do polinómio. Mais ainda, este método é bastante flexível, tornando possível modelar processos complexos para os quais não existe modelo teórico [78].

Uma desvantagem a apontar é a necessidade de um grande e denso volume de dados de modo a produzir bons modelos. Outra desvantagem que foi mencionada como vantagem é a não determinação explícita da função de regressão global em termos de fórmulas matemáticas. Por último, o ajustamento de dados a um modelo de regressão *loess* é computacionalmente intenso, o que na maioria das situações pode não ser problemático, no caso dos dados serem de dimensão excessivamente elevada pode trazer problemas [78]. É de referir que este método está sujeito ao efeito de *outliers*. Em [13] há uma versão robusta da regressão *loess* que se baseia num processo iterativo e pode ser utilizado para minimizar o efeito desses pontos.

Comandos no *software* R

Existem dois comandos no *software* R para executar o modelo de regressão *loess*, `lowess()` e `loess()`. O primeiro apenas permite o ajuste a funções polinomiais de grau 1. Pelo contrario, o comando `loess()` é mais abrangente e permite ao utilizador a escolha do grau 0, 1 ou 2 para o polinómio a ajustar. Ambos os comandos têm mais

parâmetros de entrada que podem ser definidos pelo utilizador.

3.3.3 Literatura Relacionada

Nesta secção será documentada a elevada contribuição por parte dos diferentes investigadores no desenvolvimento de métodos de NM. Fundamentalmente incidir-se-á na explicação dos métodos aplicados no estudo experimental do Capítulo 4, mas é também feita uma breve abordagem a outros trabalhos.

Os métodos de NM procuram assegurar que o viés introduzido pelo fluoróforo seja eliminado, mas que as variações biológicas intrínsecas aos dados sejam conservadas. Este viés pode-se reflectir nos efeitos espaciais e nos efeitos dependentes da intensidade.

Existem muitas técnicas de NM mas pode-se destacar a normalização global que é uma largamente utilizada. Esta técnica ajusta a média ou mediana da distribuição dos valores-M para cada *microarray* através de uma constante. Neste método de NM global, como descrito em [9], existem duas suposições distintas: *(i)* primeiro, é assumido que a quantidade total de ARN mensageiro é a mesma para as duas amostras em estudo e portanto, há aproximadamente o mesmo número de moléculas em cada; *(ii)* em segundo lugar, é assumido que os genes representados no *microarray* não são diferencialmente expressos. Muito sumariamente o processo consiste em subtrair uma constante, c, ao logaritmo da razão da expressão genética, M. Esta constante pode ser encontrada de diversas formas [10, 68]. Os métodos globais não são, contudo, apropriados quando o viés introduzido pelo fluoróforo depende da intensidade total dos pontos do *microarray*.

A estratégia mencionada no parágrafo anterior é muito ampla e, portanto, não tem em consideração variações mais específicas como os efeitos espaciais ou dependentes da intensidade. É então assim que surge a necessidade da regressão *loess* ser utilizada como ferramenta de normalização, removendo os efeitos dependentes da intensidade nos logaritmos da razão da expressão genética [68] ao nível de todo o *microarray*. Nesta abordagem a suposição é que ou os genes não são diferencialmente expressos ou os genes são influenciados por efeitos aleatórios [76]. Existem ainda outros métodos de normalização que corrigem os efeitos dependentes da intensidade [22, 36].

A maioria dos métodos de NM podem ser aplicados: *(i)* tanto a nível global, ou seja, a todo o *microarray* e portanto a todo o conjunto de dados; *(ii)* como a nível local, isto é, podem ser aplicados a subconjuntos do *microarray*, também designados por grupos-PT (Secção 1.2.1), e por isso a subconjuntos dos dados. Consequentemente, a técnica introduzida no parágrafo anterior pode ser aplicado a grupos-PT [68], o que faz com que um método com a finalidade de remover os efeitos dependentes da intensidade pode parcialmente corrigir o efeito espacial presente nos dados. Para aplicar uma estratégia de NM a diferentes regiões do *microarray* é necessário que as condições e suposições subjacentes ao método sejam válidas ao nível local.

Existem também métodos específicos para a correcção dos efeitos espaciais. Estes métodos podem basear-se na regressão *loess* bidimensional [61] ou ainda basear-se noutras teorias como métodos de filtros de medianas [73] e médias pesadas [75].

Foram ainda desenvolvidos muitos outros tipos de métodos de NM com outras suposições e com a aplicação de outras teorias matemáticas. Alguns trabalhos sobre métodos de NM fazem um bom resumo das diferentes metodologias desenvolvidas até aos dias de hoje, como por exemplo [9, 61, 62, 76].

O presente trabalho foca, não a apresentação de um método de NM em particular mas, a comparação de diferentes métodos de NM, individualmente ou em combinação, com métodos de CB. Existem já diversos trabalhos de avaliação e comparação de métodos de NM onde tanto variam os diferentes métodos utilizados como a estratégia de avaliação dos mesmos, como se explica no parágrafo seguinte.

Em [69] alguns métodos de NM são avaliados considerando o efeito na localização e escala dos logaritmos da razão da expressão genética, valores-M. Estes são ainda avaliados com base na habilidade de detectar genes dos quais é sabido que são diferencialmente expressos, através de estatísticas-t. O trabalho [39] procede com uma avaliação deste tipo de estratégias através de gráficos RIP[9]. Alguns investigadores avaliaram as técnicas de NM com base na variação entre réplicas de *microarrays* e ainda com respeito ao viés, erro quadrático médio e variância em dados simulados [48]. A identificação de genes diferencialmente expressos é utilizada muito frequentemente como forma de avaliação destes métodos, no entanto, as estatísticas envolvidas podem diferir

[9] ***R**ank **I**ntensity **P**lot*

de estudo para estudo. Assim surge [50], um trabalho que engloba várias estatísticas, seis mais concretamente, onde os genes são ordenados consoante o resultado de cada estatística. Isto permitiu aos investigadores determinar a habilidade de cada estatística em detectar genes diferencialmente expressos. De modo bastante diferente emerge [76], um estudo onde os métodos de NM são avaliados usando a capacidade preditiva de um classificador induzido de dados de *microarrays* de ADN-complementar. O classificador induzido é o k-NN e a estimativa do erro de classificação é determinada a partir do método de validação cruzada *leave one out* (LOO-CV).

3.3.4 Métodos de Normalização

Aqui focam-se os métodos de NM considerados no estudo experimental realizado no Capítulo 4, os quais podem ser obtidos com o auxílio do *software R*, no pacote *marray*, e que se baseiam no modelo de regressão *loess*. Na Tabela 3.2 encontram-se descritos os comandos desses métodos e as abreviaturas usadas de acordo com [76]. Todos eles têm

Método	Função com especificação dos parâmetros	Abreviatura
NoNorm	**maNorm**(data,norm="none")	NN
Igloess	**maNorm**(data, norm = "loess", subset = TRUE, span = 0.4)	Intensity Global loess (IG)
Illoess	**maNorm**(data, norm = "printTipLoess", subset = T, span = 0.4)	Intensity Local loess (IL)
Slloess	**maNormMain**(data, f.loc = list(maNorm2D(g = "maPrint-Tip", subset = T, span = 0.4)))	Spatial Local loess (SL)
IgloessSlloess	d=**maNorm**(data, norm = "loess", subset = TRUE, span = 0.4) / **maNormMain**(d, f.loc = list(maNorm2D(g = "maPrintTip", subset = T, span = 0.4)))	Intensity Global loess seguida de Spatial Local loess (IG-SL)
IlloessSlloess	d=**maNorm**(data, norm = "printTipLoess", subset = T, span = 0.4) / **maNormMain**(d, f.loc = list(maNorm2D(g = "maPrintTip", subset = T,span = 0.4)))	Intensity Local loess seguida de Spatial Local loess (IL-SL)

Tabela 3.2: Métodos de NM considerados no estudo experimental com indicação dos comandos usados no *software R/Bioconductor* através do pacote *marray*.

como objectivo o ajustamento dos valores-M para cada ponto (gene) do *microarray*, através da eliminação do viés introduzido pelo fluoróforo que serviu para colorir o material genético aquando da parte laboratorial.

Foram usados dois tipos de métodos de NM: *(i)* métodos de um passo, onde se aplicam apenas um método de NM aos dados (IG, IL, SL); *(ii)* métodos de dois passos onde, como o próprio nome indica, se aplicam dois métodos de NM consecutivamente (IG-SL e IL-SL).

Os métodos de NM aplicados recaem na suposição que a maioria dos genes não respondem a variações das condições experimentais e, consequentemente, é esperado que os valores-M ao longo do *microarray* sejam nulos. Os métodos de NM que se apresentam de seguida detectam os desvios ao comportamento esperado e corrigem-nos através de uma regressão polinomial localmente ponderada (*loess*). O modelo *loess* é robusto no sentido que genes diferencialmente expressos não afectam os pontos ajustados. Suponha-se que $x_k = \log_2 \sqrt{R_k G_k}$ e que $y_k = \log_2(R_k/G_k)$. O modelo *loess* produz uma estimativa, $y(x_k)$, da dependência dos $\log_2(R/G)$ em relação aos $\log_2 \sqrt{RG}$. A função de regressão é usada para, ponto por ponto, corrigir os valores $\log_2(R/G)$ medidos. Assim, nos métodos de NM é subtraído o valor ajustado pelo modelo *loess* ao valor observado M, para cada ponto do gráfico-MA. Explicam-se de seguida os métodos de NM usados.

Não Normalização (NoNorm):

Nenhuma transformação é aplicada aos valores-M.

Normalização dependente da intensidade (Igloess) [68]:

Este método transforma os valores-M em $M - c(A)$, onde $c(A)$ é o ajuste *loess* ao gráfico-MA. Esquematicamente, tem-se

$$M = \log_2(R/G) \rightarrow M' = M - c(A)$$

Normalização dentro de cada grupo-PT (Illoess) [68]:

Este método realiza uma normalização dependente da intensidade mas usando uma curva *loess* para cada grupo-PT do *microarray*. Tem-se

$$M = \log_2(R/G) \rightarrow M' = M - c_i(A)$$

onde $c_i(A)$ é o ajuste *loess* ao gráfico-MA para a i-ésimo grupo-PT, $i = 1, ..., N$, e N representa o número total de grupos-PT.

Normalização espacial 2D(Slloess) [61]:

A ideia por trás deste método é a mesma que a do método Igloess mas, neste caso, uma superfície polinomial (curva bidimensional *loess*) é ajustada aos dados, tendo em conta a sua localização espacial no *microarray*. Tem-se então

$$M = \log_2(R/G) \rightarrow M' = M - \text{loess}(l_i, c_i)$$

onde $\text{loess}(l_i, c_i)$ é uma curva bidimensional *loess* que é uma função da posição da linha, l_i, e da posição da coluna, c_i, dos pontos do *microarray* no i-ésimo grupo-PT, onde $i = 1, ..., N$, e N denota o número total de grupos-PT.

Normalização a dois passos (IgloessSlloess):

A normalização é feita em dois passos usando primeiro o método Igloess e a seguir Slloess. Assim, tem-se

$$\begin{aligned} M = \log_2(R/G) &\rightarrow M' = M - c(A) \\ &\rightarrow M'' = M' - \text{loess}(l_i, c_i) \end{aligned}$$

onde $c(A)$ é o ajuste *loess* ao gráfico-MA e $\text{loess}(l_i, c_i)$ é uma curva bidimensional *loess* que é uma função da posição da linha, l_i, e da posição da coluna, c_i, dos pontos do *microarray* no i-ésimo grupo-PT, onde $i = 1, ..., N$, e N denota o número total de grupos-PT.

Normalização a dois passos (IlloessSlloess):

A normalização é feita em dois passos usando primeiro o método Illoess e a seguir Slloess. Assim, tem-se

$$\begin{aligned} M = \log_2(R/G) &\rightarrow M' = M - c_i(A) \\ &\rightarrow M'' = M' - \text{loess}(l_i, c_i) \end{aligned}$$

onde $c_i(A)$ é o ajuste *loess* ao gráfico-MA para a i-ésimo grupo-PT, $i = 1, ..., N$, onde N representa o número total de grupos-PT e loess(l_i, c_i) é uma curva bidimensional *loess* que é uma função da posição da linha, l_i, e da posição da coluna, c_i, dos pontos do *microarray* no i-ésimo grupo-PT, onde $i = 1, ..., N$, e N denota o número total de grupos-PT.

Capítulo 4

Estudo experimental

4.1 Introdução

O estudo experimental que se segue foi realizado com a finalidade de avaliar e comparar 6 métodos de CB combinados com 6 métodos de NM, originando 36 métodos no total, aplicados a dados de *microarrays* de ADN-complementar. As 36 estratégias de pré-processamento foram avaliadas com base no desempenho preditivo de classificadores induzidos de dados de *microarrays* de três tipos de cancro. Os dados utilizados provêm de três bases de dados extraídas do repositório da Universidade de Stanford nos EUA [63]. Na Tabela 4.1 encontra-se uma breve descrição dessas bases de dados e o sítio onde as mesmas se encontram acessíveis.

A fim de avaliar os métodos de CB e de NM aplicados aos dados em estudo escolheram-se os classificadores dos k-vizinhos mais próximos e as máquinas de suporte vectorial. O primeiro, para além de ser um classificador muito estudado e utilizado em diversas áreas de aplicação, é usado em [76] para avaliar estratégias de NM. As MSV têm sido aplicadas com êxito ao problema de classificação de cancro [28], o que motivou a sua utilização neste estudo. O método escolhido para a estimação da taxa de erro dos classificadores foi a validação cruzada *leave-one-out* (LOO-CV) descrito na Secção 2.5.1. A taxa de erro LOO-CV é pouco enviesada mas pode ter elevada variabilidade [2, 47], no entanto, este método foi seleccionado em virtude de ser frequentemente utilizado em trabalhos de investigação onde o número de exemplos de treino é reduzido, como

Base de Dados	Descrição
Liver	Microarrays: 207; K=2; Classes: normal (76), tumor (131); http://genome-www5.stanford.edu/cgi-bin/publication/viewPublication.pl?pub_no=107
Lung	Microarrays: 65; K=5; Classes: normal (5); adenocarcinoma (39) (AC), squamous cell carcinoma (13) (SCC), large cell lung cancer (4) (LCLC), small cell lung cancer (4) (SCLC); http://genome-www5.stanford.edu/cgi-bin/publication/viewPublication.pl?pub_no=100
Lymphoma	Microarrays: 107; K=3; Classes: normal (30), diffuse large B-cell lymphoma (DLBCL) (68), follicular lymphoma(FL) (9); http://genome-www5.stanford.edu/cgi-bin/publication/viewPublication.pl?pub_no=79

Tabela 4.1: Tabela com informação sobre o número de *microarrays*, o número de classes (K) e a designação de cada classe para cada uma das 3 bases de dados usadas.

é o caso do presente estudo. Para além deste facto, esta metodologia foi igualmente implementada em [76], o que motivou a sua aplicação na presente dissertação.

As próximas secções dizem respeito à metodologia usada, à aplicação computacional e os resultados obtidos na avaliação dos 36 métodos de pré-processamento. O estudo experimental dividiu-se segundo duas vertentes: na Secção 4.2 tomam-se todos os genes contidos nas bases de dados e na Secção 4.3 tomam-se subconjuntos de genes altamente discriminativos obtidos usando três critérios distintos.

4.2 Avaliação dos métodos de pré-processamento

As duas próximas subsecções abordam os detalhes da implementação deste estudo experimental, os resultados e as análises. É ainda explicado o modelo aditivo que se propõe para as taxas de erro LOO-CV dos classificadores usados. Sendo o modelo válido, o objectivo é dar uma interpretação do modo como essas taxas podem ser subdivididas em termos da contribuição dada pelos métodos de CB e de NM aplicados aos dados. Da mesma forma o modelo aditivo é aplicado para interpretar a taxa de redução do erro LOO-CV proporcionada pela aplicação de um método de pré-processamento particular. O estudo comparativo realizado na presente dissertação é apenas baseado em diferentes resultados exploratórios.

4.2.1 Detalhes da implementação

Da aplicação combinada de 6 métodos de CB, identificados na Tabela 3.1, com 6 métodos de NM, mencionados na Tabela 3.2, obtiveram-se 36 estratégias de pré-processamento. Cada par de estratégias $(\mathsf{CB} = i, \mathsf{NM} = j), i = \{\mathsf{NB}, \mathsf{sub}, \mathsf{half}, \mathsf{min}, \mathsf{edw}, \mathsf{nexp}\}$ e $j = \{\mathsf{NN}, \mathsf{IG}, \mathsf{IL}, \mathsf{SL}, \mathsf{IG}-\mathsf{SL}, \mathsf{IL}-\mathsf{SL}\}$, resulta de uma combinação de um método de CB com um de NM.

A metodologia usada para transformar cada base de dados original em 36 conjuntos de dados pré-processados é muito semelhante à proposta por [76]. Contudo, ao contrário do trabalho [76] neste estudo são também aplicados os métodos de CB. Esquematicamente, a Figura 4.1 ilustra o processo implementado sobre cada base de dados. Independentemente do número de *microarrays*, são aplicadas 36 estratégias combinadas de CB e NM que formam os 36 pares $(\mathsf{CB} = i, \mathsf{NM} = j), i = \{\mathsf{NB}, \mathsf{sub}, \mathsf{half}, \mathsf{min}, \mathsf{edw}, \mathsf{nexp}\}$ e $j = \{\mathsf{NN}, \mathsf{IG}, \mathsf{IL}, \mathsf{SL}, \mathsf{IG}-\mathsf{SL}, \mathsf{IL}-\mathsf{SL}\}$. Em primeiro lugar, é aplicada uma técnica de CB e posteriormente uma de NM. O par (NB, NN)[1] serve de linha de base e refere-se à inexistência de qualquer tipo de pré-processamento aplicado, ou seja, nem CB nem NM.

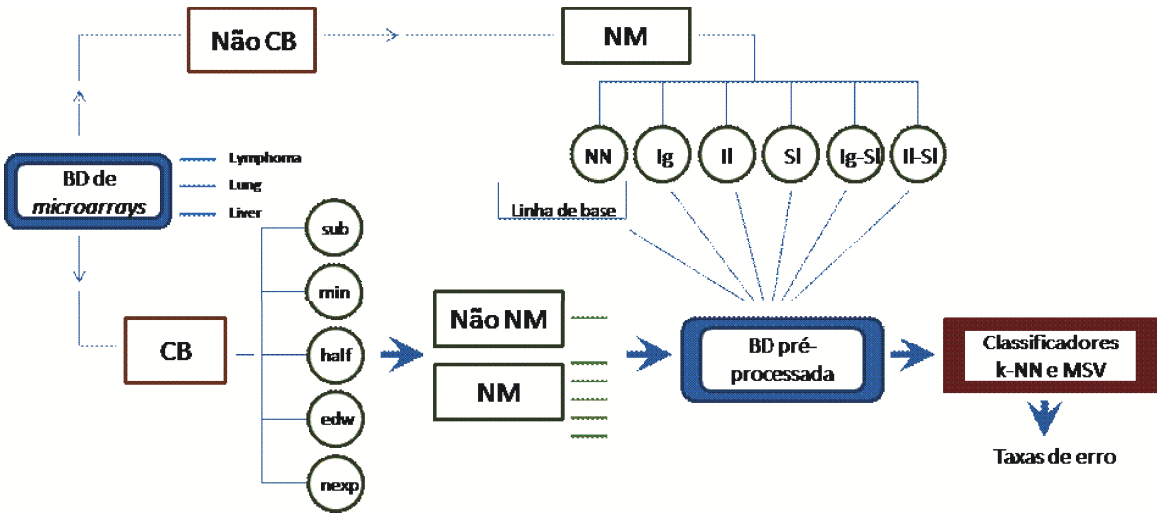


Figura 4.1: Diagrama da implementação dos métodos de CB e NM sobre uma base de dados de *microarrays*. Especificação da tarefa de classificação para os classificadores k-NN e MSV.

Cada conjunto de dados pré-processado $(\mathsf{CB} = i, \mathsf{NM} = j), i = \{\mathsf{NB}, \mathsf{sub}, \mathsf{half}, \mathsf{min}, \mathsf{edw}, \mathsf{nexp}\}$

[1] No par (NB, NN), NB representa Não correcção de *Background* e NN representa Não Normalização.

e $j = \{\mathsf{NN}, \mathsf{IG}, \mathsf{IL}, \mathsf{SL}, \mathsf{IG}-\mathsf{SL}, \mathsf{IL}-\mathsf{SL}\}$, para qualquer uma das três base de dados, foi obtido da seguinte forma:

(i) para cada ponto do *microarray* um método de CB é aplicado para corrigir os valores das intensidades medidas verde e vermelha, R e G;
(ii) os valores-M são calculados (utilizando as estimativas médias das intensidades vermelha e verde de *foreground* e *background*, R_f, R_b, G_f, G_b) e posteriormente normalizados de acordo com uma estratégia de NM;
(iii) apenas as sondas que estão presentes em todos os *microarrays* são utilizadas para análise. Sempre que são encontrados diversos valores para a mesma sonda faz-se uma média dos valores-M, continuando assim com apenas um valor-M para cada sonda;
(iv) os valores-M resultantes são centrados e reduzidos à norma unitária;
(v) os valores omissos são imputados (usando a função `pamr.knnimpute` [64] no *software* R). Como resultado, o número m de amostras de tecidos (*i.e. microarrays*) e o número n de atributos (genes) a serem considerados são: $m = 107$ e $n = 7079$ para a base de dados Lymphoma; $m = 207$ e $n = 21901$ para a Liver e $m = 65$ e $n = 22646$ para a Lung; cfr. Tabela 4.1;
(vi) à matriz $m \times n$ é adicionada uma coluna com as classes, indicadas na Tabela 4.1, de cada *microarray* constituinte da base de dados.

O procedimento LOO-CV para estimar a taxa de erro para o classificador k-NN é implementado no *software* R da mesma forma que [76]. Este procedimento foi já explicitado na Secção 2.5.1 com um diagrama exemplificativo. No entanto, na implementação computacional do classificador k-NN, usam-se dois procedimentos LOO-CV. O primeiro tem a finalidade de determinar a taxa de erro associado ao classificador k-NN, mas o segundo tem como objectivo calcular o valor óptimo, k^*, do número de vizinhos mais próximos para cada nova instância a ser classificada.

Como a Figura 4.2 ilustra, o conjunto das m instâncias é particionado, primeiramente, em dois subconjuntos, um de teste composto pela instância *abc* e outro de treino com os restantes $m-1$ objectos. Dado um valor de k específico, o classificador k-NN é utilizado para predizer a classe do exemplo de teste *abc* tendo em conta os $m-1$ exemplos de treino. Este procedimento é repetido até que todas as instâncias do conjunto original de dados tenham sido usadas como instâncias de teste. O problema que se coloca é

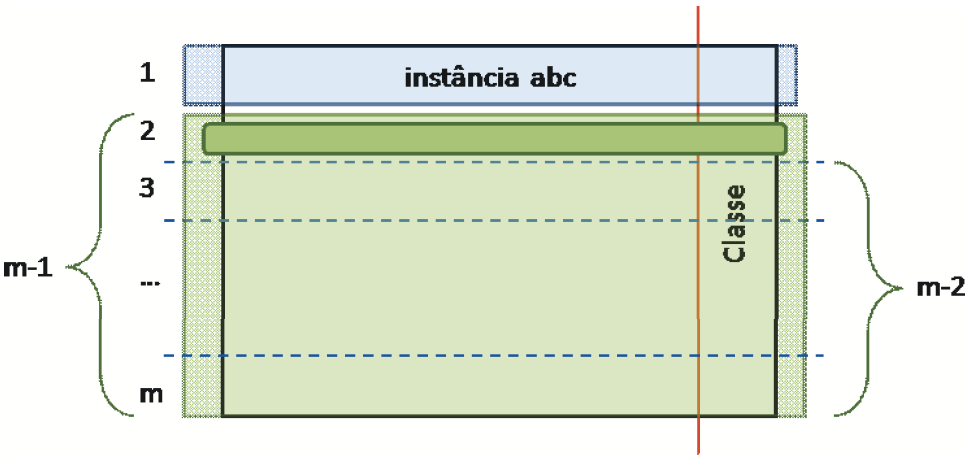


Figura 4.2: Representação esquemática do procedimento LOO-CV para a escolha do k^*.

qual o valor de k a ser escolhido para a classificação de cada exemplo de teste?. Para determinar o valor óptimo, k^*, os restantes $m-1$ exemplos do conjunto de treino são usados para determinar as estimativas do erro LOO-CV para valores de $k = \{3, \ldots 10\}$. Assim, para cada valor de k e para cada instância do conjunto de treino, em cada passo, um exemplo é deixado de fora e classificado usando um classificador k-NN induzido dos restantes $m-2$ exemplos. Como resultado é escolhido aquele valor de k que produz uma menor taxa de erro LOO-CV.

Depois de ter sido determinado o valor de k^* o processo é repetido tal que todas as instâncias do conjunto original de dados tenham sido usadas como exemplos de teste. A taxa de erro final do conjunto original de dados é calculada pela média das taxas de erro para cada uma das m iterações. Para realizar este procedimento foram utilizadas as funções `class/knn.cv(k = 3,...,10)` e `class/knn(k = k*)` no *software R*.

Para determinar a taxa de erro LOO-CV para o classificador MSV foi usado o *software* RapidMiner onde foi aplicado o operador de validação cruzada em conjunto com o operadorLibSVMLearner. Este último foi implementado com um *kernel* linear que implementa o pacote libsvm [12]. Apresenta-se na Figura 4.3, de forma figurativa, um projecto em RapidMiner onde é aplicado o operador LibSVMLearner com validação cruzada.

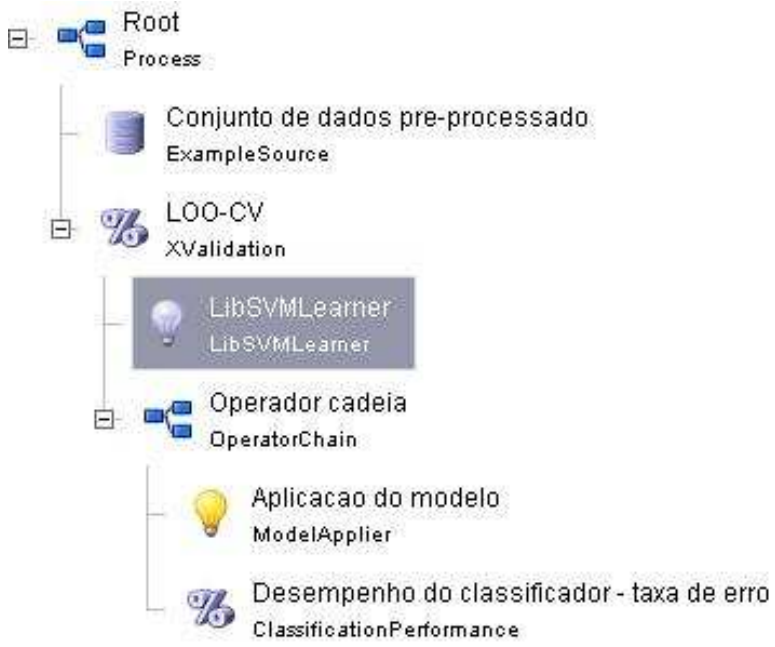


Figura 4.3: Representação de um projecto base no *software* RapidMiner. Especificação do procedimento de validação cruzada e do operador LibSVMLerner que implementa o pacote libsvm [12].

4.2.2 Resultados e discussão

A Tabela 4.2 mostra as taxas de erro LOO-CV para as 36 estratégias $(\mathsf{CB} = i, \mathsf{NM} = j), i = \{\mathsf{NB}, \mathsf{sub}, \mathsf{half}, \mathsf{min}, \mathsf{edw}, \mathsf{nexp}\}$ e $j = \{\mathsf{NN}, \mathsf{IG}, \mathsf{IL}, \mathsf{SL}, \mathsf{IG}-\mathsf{SL}, \mathsf{IL}-\mathsf{SL}\}$, avaliadas para cada base de dados. Globalmente os resultados retratam menores taxas de erro para o classificador MSV.

Com vista a examinar os eventuais melhoramentos do desempenho preditivo de cada classificador sobre os conjuntos de dados pré-processados, e assim dar significado ao efeito da melhoria devido à aplicação de um método de CB ou de NM, cada um deles separadamente, ou ainda devido à interacção entre ambos os métodos, CB↔NM, assume-se um modelo aditivo de dois factores para as taxas de erro LOO-CV. Assim, dados $S_b = \{\mathsf{NB}, \mathsf{sub}, \mathsf{half}, \mathsf{min}, \mathsf{edw}, \mathsf{nexp}\}$ e $S_n = \{\mathsf{NN}, \mathsf{IG}, \mathsf{IL}, \mathsf{SL}, \mathsf{IG}-\mathsf{SL}, \mathsf{IL}-\mathsf{SL}\}$, estabelece-se que, a menos de um erro aleatório de média nula,

$$\begin{aligned} e(i,j) &= e(\cdot,\cdot) + \Big(e(i,\cdot) - e(\cdot,\cdot)\Big) + \Big(e(\cdot,j) - e(\cdot,\cdot)\Big) + \Big(e(i,j) - e(i,\cdot) - e(\cdot,j) + e(\cdot,\cdot)\Big) \\ &= \text{efeito global} + \text{efeito } \mathsf{CB} + \text{efeito } \mathsf{NM} + \text{efeito da interacção } \mathsf{CB}{\leftrightarrow}\mathsf{NM} \end{aligned} \quad (4.1)$$

Base de dados	Método	k-NN					
	CB	NB	sub	half	min	edw	nexp
Lymphoma	NN	26.85	21.20	23.14	21.29	25.92	19.44
	IG	12.96	16.66	20.37	16.66	22.22	11.1
	IL	12.03	15.74	17.59	15.74	21.29	11.1
	SL	16.66	18.51	17.59	18.51	20.37	12.96
	IG-SL	12.03	16.6	20.37	15.74	21.29	9.25
	IL-SL	9.25	16.6	17.59	16.6	21.29	11.11
Lung	NN	35.38	26.15	36.92	36.92	35.38	35.38
	IG	23.07	29.23	41.53	41.53	41.53	32.3
	IL	20	27.69	38.46	38.46	41.53	29.23
	SL	32.3	27.69	29.23	29.23	36.92	30.76
	IG-SL	24.61	30.76	35.38	36.92	44.61	29.23
	IL-SL	21.53	24.61	41.53	41.53	35.38	27.69
Liver	NN	16.9	16.42	17.39	17.39	17.87	16.90
	IG	13.52	15.94	14.97	12.56	11.59	13.04
	IL	13.04	11.59	15.94	15.94	14.00	13.52
	SL	17.39	14.49	14.00	14.00	16.42	14.49
	IG-SL	19.80	11.59	14.97	15.94	15.94	10.14
	IL-SL	15.45	12.07	14.00	14.00	14.00	9.66
Base de dados	Método	MSV					
	CB	NB	sub	half	min	edw	nexp
Lymphoma	NN	16.67	14.81	14.91	14.81	14.81	13.89
	IG	7.41	5.56	12.96	5.56	11.11	7.41
	IL	5.56	5.56	12.96	5.56	9.26	7.41
	SL	12.96	6.48	12.96	6.48	12.04	9.26
	IG-SL	5.56	6.48	12.96	6.48	11.11	8.33
	IL-SL	5.56	6.48	11.11	6.48	10.19	6.48
Lung	NN	29.23	23.08	32.31	32.31	23.08	27.69
	IG	21.54	20	27.69	26.15	23.08	21.54
	IL	21.54	18.46	24.62	24.62	23.08	21.54
	SL	23.08	18.46	24.62	24.62	23.08	23.08
	IG-SL	20	18.46	24.62	26.15	23.08	21.54
	IL-SL	20	18.46	24.62	24.62	23.08	21.54
Liver	NN	3.86	3.86	3.86	3.86	3.86	3.86
	IG	3.86	3.38	3.38	3.86	3.38	3.38
	IL	4.83	3.86	3.86	3.86	4.35	3.38
	SL	4.83	4.35	4.35	4.35	3.86	3.38
	IG-SL	4.35	3.86	3.86	4.35	3.86	3.38
	IL-SL	4.83	3.86	3.86	3.86	3.86	2.90

Tabela 4.2: Taxas de erro LOO-CV (%), agrupadas por base de dados, para as 36 estratégias (CB, NM).

onde $e(i, j)$ representa a taxa de erro para $(\mathsf{CB} = i, \mathsf{NM} = j)$ com $i \in S_b$, $j \in S_n$, sendo

$e(i,\cdot) = \sum_{j \in S_n} e(i,j)/6$ e $e(\cdot,j) = \sum_{i \in S_b} e(i,j)/6$.

No sentido de extrair mais informações da análise do efeito de CB, foram calculadas medidas exploratórias baseadas na diferença dos valores $e(\mathsf{NB}, j) - e(i,j)$, $\forall j \in S_n$ para cada método particular de CB, $i \in S_b$. Os resultados obtidos estão sumariados na Tabela 4.3 para cada método de CB. No total são calculadas 18 taxa de erros para cada método de CB (6 métodos de NM para 3 bases de dados). Assim para cada linha da Tabela 4.3 a toma das três diferenças dá um total de 18. As iniciais p, n, e, representam as diferenças positivas ($e(\mathsf{NB}, j) > e(i,j)$), negativas ($e(\mathsf{NB}, j) < e(i,j)$) e nulas ($e(\mathsf{NB}, j) = e(i,j)$), observadas para cada $j \in S_n$ respectivamente.

Deste modo, tem interesse analisar qual o método de CB que apresenta um maior número de diferenças positivas, *i.e.*, qual o método que permite obter uma maior redução (em termos absolutos) da taxa de erro. Dos resultados da Tabela 4.3 observa-se que em termos de ganho de desempenho para o k-NN o método de CB com mais diferenças positivas é o nexp (10) seguido de sub (6) e min (6), enquanto que para as MSV é o sub (14) seguido de nexp (8) e edw (8), que mostram um empate. Assim, considerando as três bases de dados, com respeito às MSV, o método sub parece ser o melhor método de CB em termos de ganhos de desempenho, seguido de nexp e edw. Em relação ao classificador k-NN observam-se resultados semelhantes para sub e nexp.

Os mesmos cálculos foram feitos relativamente à NM, todavia, as conclusões dos resultados são pouco interessantes, veja-se a Tabela 4.3. Para o classificador k-NN as diferenças positivas são para todos os métodos praticamente iguais e para o classificador MSV a situação é mesma com a excepção do método IG que apresenta mais 3 diferenças positivas que os restantes.

Os gráficos da Figura 4.4 foram executados para ilustrar a contribuição do efeito da aplicação dos métodos de CB e de NM. Portanto, estão representados nesses gráficos as diferenças, $e(i,\cdot) - e(\cdot,\cdot)$ e $e(\cdot,j) - e(\cdot,\cdot)$, para $i \in S_b$ e $j \in S_n$.

A diferença $e(i,\cdot) - e(\cdot,\cdot)$ ilustra o efeito que a correcção de *background* tem na taxa de erro global $e(i,j)$, que se decompõe da forma indicada pela Expressão 4.1. Nos gráficos da Figura 4.4(a) e 4.4(b) é possível observar que, tanto para cada de base de dados como a nível médio, os métodos de CB sub e nexp são os que têm uma contribuição para o desempenho do classificador no sentido da redução da taxa de erro $e(i,j)$. No

		p/n/e	
	Métodos	k-NN	MSV
C. de *background*	sub	8/10/10	14/2/2
	half	5/13/0	6/10/2
	min	6/12/0	7/8/3
	edw	5/12/1	8/8/2
	nexp	10/6/2	8/5/5
Normalização	IG	14/4/0	15/0/3
	IL	14/4/0	12/2/4
	SL	15/3/0	12/5/1
	IG-SL	14/3/1	12/2/4
	IL-SL	15/2/1	12/1/5

Tabela 4.3: Diferença dos valores $e(\mathsf{NB}, j) - e(i, j)$, $\forall j \in S_n$ para um método particular de $\mathsf{CB} = i \in S_b$ e diferença dos valores $e(i, \mathsf{NN}) - e(i, j)$, $\forall i \in S_b$ para um método particular de $\mathsf{NM} = j \in S_b$ para os dois classificadores. As iniciais p, n, e, representam positivos, negativos, empate, respectivamente.

caso da normalização, para o classificador MSV, Figura 4.4(d), o único método de NM com contribuição positiva, ou seja, com tendência a aumentar a taxa de erro $e(i, j)$ em termos médios é SL. Os métodos de NM com uma maior contribuição para a redução de $e(i, j)$, em termos médios, são os métodos IL-SL e IL. Para o classificador k-NN, em termos médios, são os métodos IL-SL e IL que apresentam uma contribuição no sentido da redução da taxa de erro total $e(i, j)$.

Definam-se as *taxas de redução* (TR) como medidas quantitativas para representar o ganho de desempenho com a aplicação de uma estratégia específica ($\mathsf{CB} = i, \mathsf{NM} = j), i \in S_b$ e $j \in S_n$, em relação ao caso de base (NB, NN). As TR associadas ao modelo aditivo proposto permitem quantificar a contribuição dos métodos de CB e NM no desempenho preditivo dos classificadores induzidos a partir de cada base de dados de *microarrays*. Concretamente, para a combinação ($\mathsf{CB} = i$, $\mathsf{NM} = j$), a taxa de redução

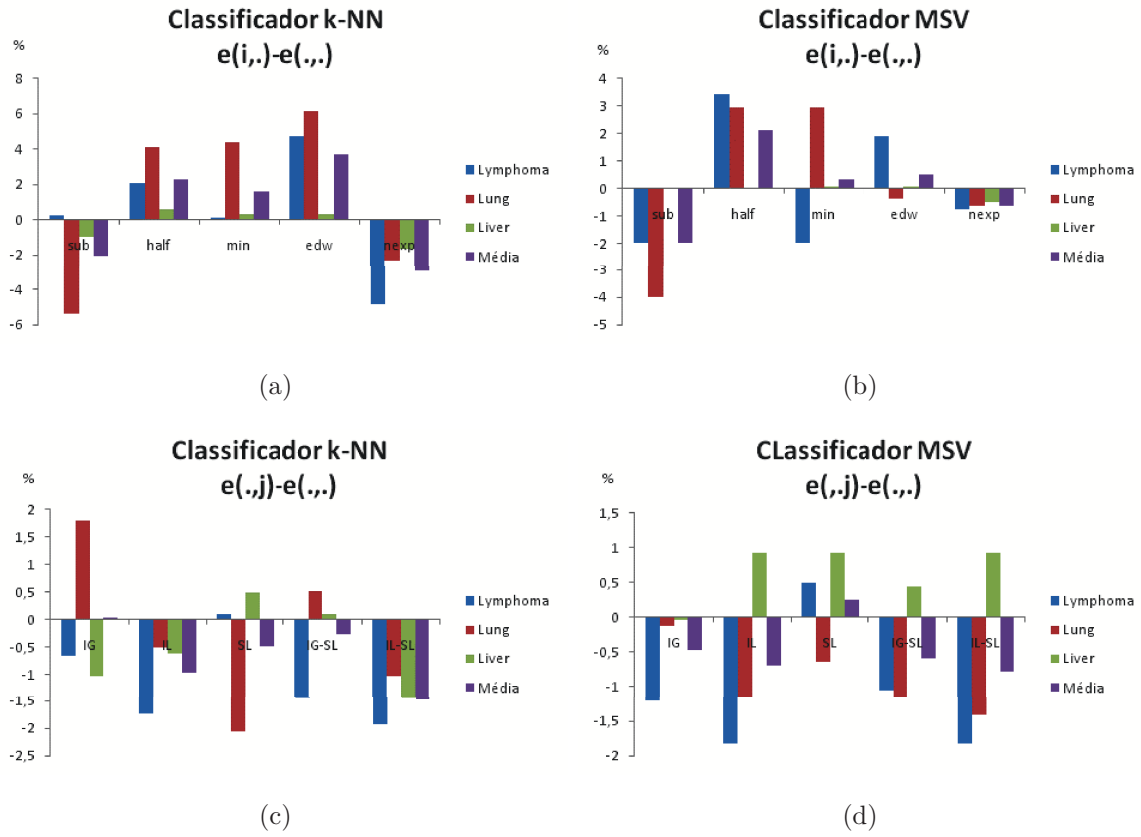


Figura 4.4: Gráficos de barras das taxas de erro relativas à contribuição de métodos de CB e NM para os classificadores k-NN e MSV, por base de dados e ainda pela média das três bases de dados.(a) Diferenças $e(i,\cdot) - e(\cdot,\cdot)$, $i \in S_b$, obtidas do classificador k-NN. (b) Diferenças $e(i,\cdot) - e(\cdot,\cdot)$, $i \in S_b$, obtidas do classificador MSV. (c) Diferenças $e(\cdot,j) - e(\cdot,\cdot)$, $j \in S_n$, obtidas do classificador k-NN. (d) Diferenças $e(\cdot,j) - e(\cdot,\cdot)$, $j \in S_n$, obtidas do classificador MSV.

é dada por,

$$\begin{aligned} TR(i,j) &= \frac{e(\mathsf{NB},\mathsf{NN}) - e(i,j)}{e(\mathsf{NB},\mathsf{NN})} \\ &= \frac{e(\mathsf{NB},\mathsf{NN}) - e(\cdot,j)}{e(\mathsf{NB},\mathsf{NN})} + \frac{e(\mathsf{NB},\mathsf{NN}) - e(i,\cdot)}{e(\mathsf{NB},\mathsf{NN})} - \\ &\qquad - \frac{e(\mathsf{NB},\mathsf{NN}) - (e(\cdot,j) + e(i,\cdot) - e(i,j))}{e(\mathsf{NB},\mathsf{NN})} \\ &= TR(i,\cdot) + TR(\cdot,j) - TR(i \leftrightarrow j) \quad i \in S_b\,, j \in S_n \end{aligned}$$

Assim, a taxa de redução $TR(i,j)$ é decomposta em três partes:
(i) uma devido à CB, $TR(i,\cdot)$;

(ii) uma devido à NM, $TR(\cdot, j)$;
(iii) uma devido à interacção dos métodos de CB e de NM, $TR(i \leftrightarrow j)$.

As *taxas de redução médias* (TRM) foram obtidas agrupando os resultados das três bases de dados em estudo. A $TRM(i, j)$ para o par $(\mathsf{CB} = i, \mathsf{NM} = j), i \in S_b$ e $j \in S_n$ é a média das três $TR(i, j)$ para esse mesmo par. De modo análogo obtiveram-se as TRM relativas aos métodos de CB, $TRM(i, \cdot)$, as TRM relativas aos métodos de NM, $TRM(\cdot, j)$, e relativas à interacção dos métodos de CB e de NM, $TRM(i \leftrightarrow j)$, para todo o $i \in S_b$ e $j \in S_n$. A Figura 4.5 ilustra as TRM para cada combinação de métodos $(\mathsf{CB} = i, \mathsf{NM} = j), i \in S_b$ e $j \in S_n$, para os dois classificadores induzidos. Nas Figuras 4.5(a) e 4.5(b) as TRM estão representadas de modo a dar ênfase aos métodos de CB e as Figuras 4.5(c) e 4.5(d) dão ênfase às TRM para cada método de NM.

A fim de se determinarem os métodos que conduziram a uma maior taxa de redução analisaram-se, por classificador, as barras verticais dos vários gráficos. Averiguando os gráficos das Figuras 4.5(a) e 4.5(b) observa-se que, na generalidade, as combinações de métodos $(\mathsf{CB} = i, \mathsf{NM} = j)$, $i \in \{\mathsf{sub}, \mathsf{nexp}\}$, $j \in S_n$ mostram uma TRM mais elevada. As barras horizontais exprimem que os métodos de CB que mostram uma maior TRM são sub, nexp e min, indicando assim uma melhor contribuição destes métodos no desempenho preditivo dos classificadores. Estes resultados parecem ser consistentes com os anteriormente apresentados na Tabela 4.3 relativa às diferenças dos erros e ainda com os gráficos da Figura 4.4.

Analisando as barras verticais, das Figuras 4.5(c) e 4.5(d), não se evidencia nenhum método de NM que proporcione, na globalidade, maiores TRM para os dois classificadores induzidos dos dados. Poder-se-á dizer que o método SL parece traduzir uma menor redução, em particular, sobre o classificador MSV. De facto, observando as barras horizontais que traduzem a contribuição dos métodos de NM segundo o modelo aditivo assumido, regista-se o menor valor observado para as barras horizontais correspondentes ao método SL para ambos os classificadores. Dos métodos de NM dá-se destaque ao facto do método de 2-passos IL-SL produzir as duas maiores TRM (primeira maior para o k-NN e segunda maior para o MSV, nas barras horizontais). De novo, estes resultados parecem ser consistentes com os já apresentados.

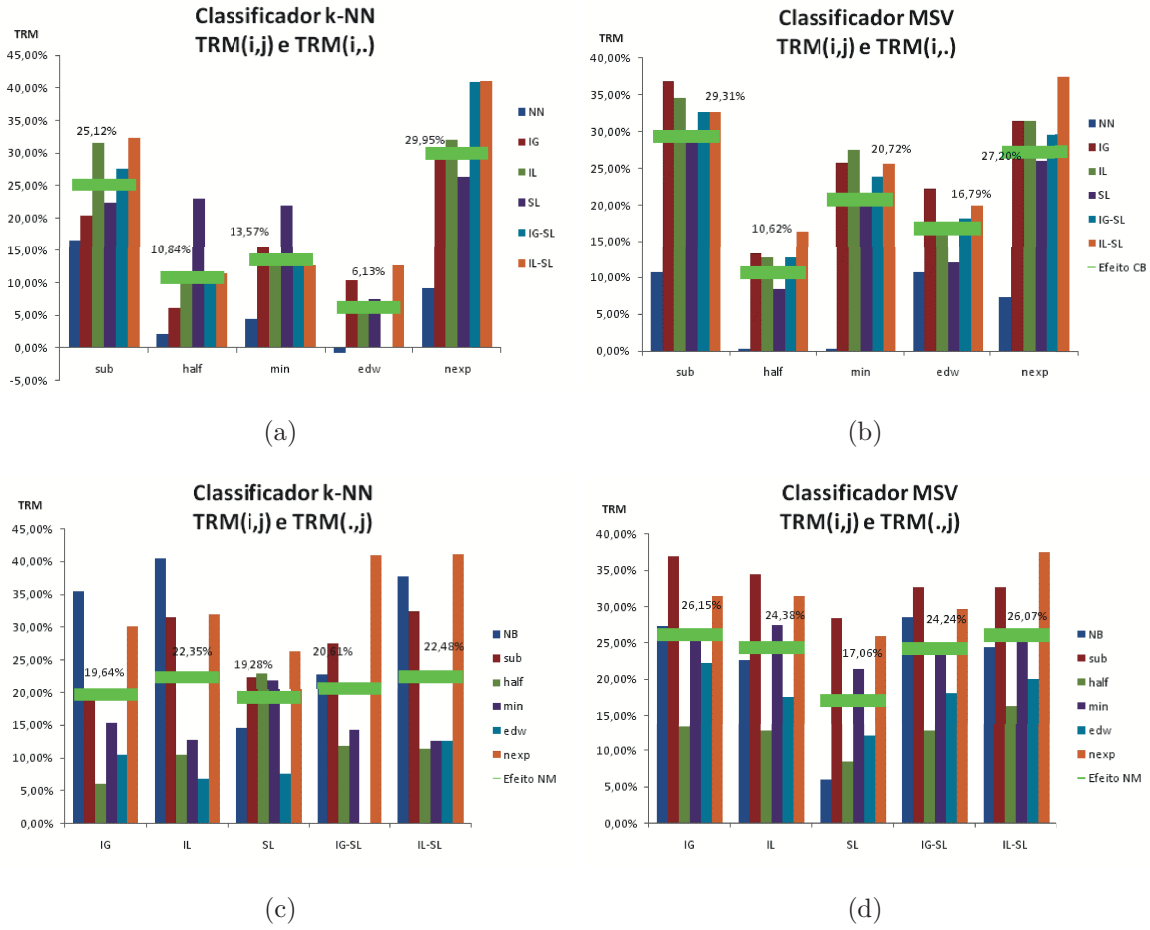


Figura 4.5: Gráficos de barras das $TRM(i,j)$, $i \in S_b$ e $j \in S_n$, obtidas do classificador k-NN (gráficos (a) e (c)) e obtidas do classificador MSV (gráficos (b) e (d)). As barras verticais representam as TRM de cada par (CB,NM) agrupadas por métodos de CB (gráficos (a) e (b)) e por métodos de NM (gráficos (c) e (d)). As barras horizontais representam as TRM para cada método de CB nos gráficos (a) e (b) e para cada método de NM nos gráficos (c) e (d).

4.3 Selecção de genes. Um estudo de caso.

O ruído inerente aos dados de *microarrays* pode ser categorizado em ruído técnico e biológico [72]. No Capítulo 3 foram abordadas técnicas desenvolvidas por diversos investigadores com a finalidade de remover as fontes de ruído técnico através da implementação de métodos de CB e de NM. O ruído biológico é adicionado aos dados através dos próprios genes em estudo, ou seja, é introduzido pelos genes que não apresentam relevância para a tarefa de classificação. Em virtude do número de amostras

ser reduzido comparativamente com o número de genes que é excessivamente elevado, a presença de ruído biológico pode afectar negativamente a precisão da classificação.

Nesta secção é abordado o problema da selecção de genes, *i.e.* a identificação de um conjunto reduzido de genes altamente discriminativos [28]. O principal objectivo é avaliar até que ponto a remoção de ruído técnico através de métodos de CB e de NM influencia o processo de selecção de genes. Para este fim, foi conduzido um estudo de caso com a base de dados Lymphoma usando apenas o classificador MSV com um *kernel* linear. Diferentes estratégias de selecção de atributos foram aplicadas aos conjuntos de dados onde previamente foram implementados 6 métodos de CB (NB, sub, half, min, edw e nexp) em combinação com dois métodos de NM de 2-passos (IG-SL e IL-SL), veja-se a Figura 4.6. O objectivo deste estudo foi alargar a análise comparativa

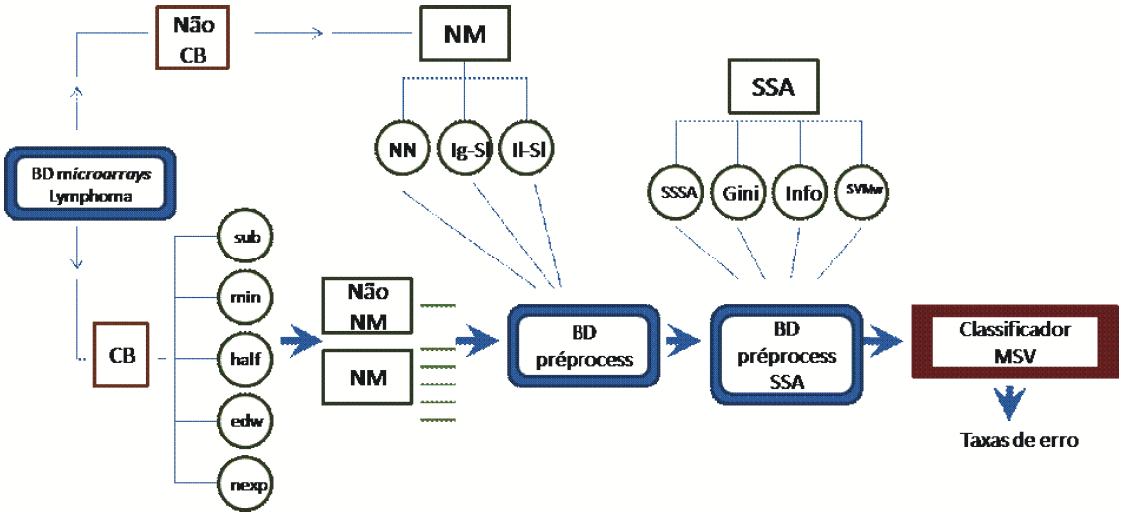


Figura 4.6: Diagrama da implementação dos métodos de CB, NM e SSA sobre a base de dados Lymphoma. Especificação da tarefa de classificação para o classificador MSV.

das diferentes técnicas de CB e NM para avaliar o contributo destes no contexto do problema de selecção de genes.

4.3.1 Detalhes da implementação

O problema de selecção de genes na classificação de cancro é um caso particular de um problema muito investigado na área da aprendizagem automática que é conhecido como selecção de subconjuntos de atributos (SSA)[2] [47]. O objectivo é seleccionar um subconjunto de atributos que produz o melhor desempenho na tarefa de classificação. Neste estudo em particular os atributos referem-se a genes. No *software* RapidMiner [44] há uma variedade de técnicas de SSA mas, para este trabalho, foram somente escolhidas aquelas que conseguiram identificar um melhor subconjunto de genes discriminativos a um menor custo computacional. A estratégia escolhida é composta por dois passos.

No primeiro passo, um esquema de pesos é usado para ordenar os genes relativamente ao seu poder discriminativo. Neste estudo comparam-se três esquemas de pesos implementados no RapidMiner. Os esquemas com os operadores InfoGainWeighting e GiniIndexWeighting determinam a relevância dos genes através do cálculo de duas medidas comummente utilizadas na indução de árvores de decisão, o ganho de informação e o índice de impureza Gini (mais informação sobre estas medidas pode ser encontrada, por exemplo, em [51]). O terceiro operador SVMWeighting usa os coeficientes do classificador MSV como pesos para os atributos. De seguida é aplicado um filtro (com um determinado critério definido pelo utilizador) de modo a seleccionar, para análises posteriores, apenas aqueles atributos com um peso superior a um determinado limiar.

No segundo passo, a selecção de atributos (um problema de optimização discreta) é realizada usando um algoritmo guloso (*greedy*) com uma estratégia de *forward selection*. O algoritmo começa com o conjunto de atributos vazio e vai adicionando novos atributos que trazem um maior decréscimo à taxa de erro LOO-CV. O processo termina quando a adição de um novo atributo não traz nenhum melhoramento no desempenho do classificador induzido.

Após estes dois passos é obtido o subconjunto de atributos final. Assim, o objectivo da estratégia descrita é conseguir um subconjunto do conjunto original de atributos muito mais reduzido, mas que ao mesmo tempo os classificadores induzidos apresentem uma taxa de erro inferior, ou pelo menos que nunca seja superior, à obtida com o conjunto

[2]Tradução do termo inglês *feature subset selection*, em abreviatura FSS.

completo de atributos.

A presente dissertação baseou-se no trabalho de [28] onde é apresentado um estudo sobre selecção de genes usando o classificador MSV. Nesse trabalho foi proposto e avaliado o esquema de peso SVMWeighting que está implementado no RapidMiner. Os resultados do desempenho foram avaliados usando MSVs. Esses resultados mostraram que a ordenação de genes, como consequência dos pesos de uma MSV com *kernel* linear, permitiu encontrar apenas dois genes da base de dados em estudo que originaram uma taxa de erro LOO-CV de zero. Como observação, refere-se que de forma semelhante ao estudo proposto por [28], na presente implementação apenas é realizado um ciclo interno de validação cruzada (ver Figura 4.7 B). Esta estratégia fornece, no entanto, uma estimativa optimista do erro de classificação. Como argumentado em [2], a taxa de erro LOO-CV interna não considera o "viés de selecção"[3]. Deste modo, deveria ser implementado uma validação cruzada externa, ver Figura 4.7 C. Em [2] é explicado que se um conjunto de teste é usado para estimar o erro de classificação haverá um viés de selecção se esse mesmo conjunto é usado no processo de selecção de genes. Para que se obtenha uma estimativa não enviesada é necessário que o conjunto de teste não tenha nenhum papel no processo de SSA. Uma vez que o foco deste estudo baseia-se na avaliação do impacto que a correcção de *background* e a normalização têm na selecção de genes, e devido ao esforço computacional adicional preciso para implementar tal esquema de validação cruzada externa, utilizou-se apenas o uso do procedimento interno LOO-CV.

4.3.2 Resultados e discussão

Para explorar o efeito da SSA estendeu-se o modelo aditivo introduzido na subsecção anterior considerando agora três factores, CB, NM e SSA, definindo-se de forma semelhante novas TR. Assim, assume-se que a taxa de erro $e(i, j, k)$ do classificador MSV induzido da base de dados Lymphoma, para os quais foram aplicados o método $\mathsf{CB} = i$ com $i \in \{\mathsf{NB}, \mathsf{sub}, \mathsf{half}, \mathsf{min}, \mathsf{edw}, \mathsf{nexp}\}$, o método de $\mathsf{NM} = j$ com $j \in \{\mathsf{NN}, \mathsf{IG}-\mathsf{SL}, \mathsf{IL}-\mathsf{SL}\}$ e o procedimento $\mathsf{SSA} = k$ com $k \in \{\mathsf{SSSA}, \mathsf{InfoGain}, \mathsf{GiniIndex}, \mathsf{SVMWeighting}\}$, satisfaz

[3]Tradução do termo inglês *selection bias*.

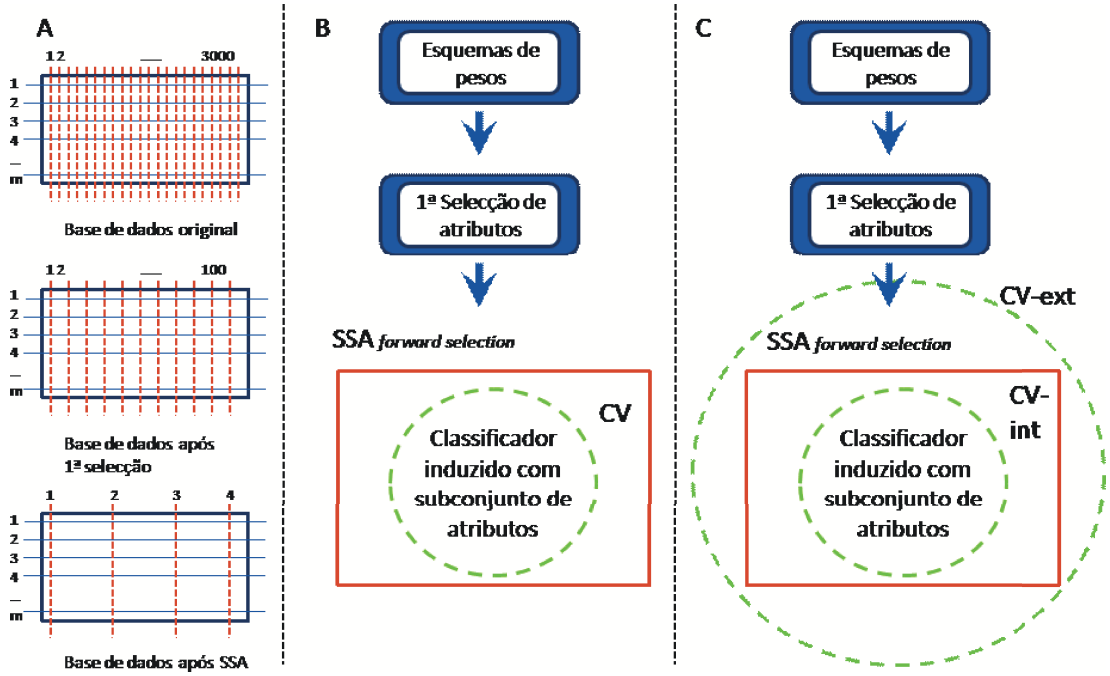


Figura 4.7: **A.** Representação da evolução da matriz $m \times n$, onde m representa o número de tecidos e n o número de genes, à medida que os procedimentos de selecção de genes são aplicados (os números 4, 100 e 3000 são números aleatórios apenas usados para ilustração do processo). **B.** Representação da implementação realizada do processo de SSA apenas com validação cruzada interna. **C.** Representação da implementação do processo de SSA com procedimento de validação cruzada interna e externa.

o seguinte modelo,

$$e(i,j,k) = e(\cdot,\cdot,\cdot) + \Big(e(i,\cdot,\cdot) - e(\cdot,\cdot,\cdot)\Big) + \Big(e(\cdot,j,\cdot) - e(\cdot,\cdot,\cdot)\Big) + \Big(e(\cdot,\cdot,k) - e(\cdot,\cdot,\cdot)\Big) +$$
$$\Big(e(i,j,\cdot) - e(i,\cdot,\cdot) - e(\cdot,j,\cdot) + e(\cdot,\cdot,\cdot)\Big) + \Big(e(i,\cdot,k) - e(i,\cdot,\cdot) - e(\cdot,\cdot,k) + e(\cdot,\cdot,\cdot)\Big) +$$
$$\Big(e(\cdot,j,k) - e(\cdot,j,\cdot) - e(\cdot,\cdot,k) + e(\cdot,\cdot,\cdot)\Big) +$$
$$\Big(e(i,j,k) + e(i,\cdot,\cdot) + e(\cdot,j,\cdot) + e(\cdot,\cdot,k) - e(i,j,\cdot) - e(i,\cdot,k) - e(\cdot,j,k) - e(\cdot,\cdot,\cdot)\Big)$$

$$= \text{efeito global} + \text{efeito CB} + \text{efeito NM} + \text{efeito SSA} + \text{efeito da interacção CB} \leftrightarrow \text{NM} +$$
$$\text{efeito da interacção CB} \leftrightarrow \text{SSA} + \text{efeito da interacção NM} \leftrightarrow \text{SSA} +$$
$$\text{efeito da interacção CB} \leftrightarrow \text{NM} \leftrightarrow \text{SSA}$$

onde $e(i,j,k)$ representa a taxa de erro para $\text{CB} = i$, $\text{NM} = j$ e $\text{SSA} = k$, $i \in S_b = \{\text{NB}, \text{sub}, \text{half}, \text{min}, \text{edw}, \text{nexp}\}$, $j \in S_n = \{\text{NN}, \text{IG}-\text{SL}, \text{IL}-\text{SL}\}$, $k \in S_s =$

{SSSA, InfoGain, GiniIndex, SVMWeighting} e $e(i,\cdot,\cdot) = \sum_{j\in S_n}\sum_{k\in S_s} e(i,j,k)/(j\times k)$, $e(\cdot,j,\cdot) = \sum_{i\in S_b}\sum_{k\in S_s} e(i,j,k)/(i\times k)$, $e(\cdot,\cdot,k) = \sum_{i\in S_n}\sum_{j\in S_s} e(i,j,k)/(i\times j)$.

A Tabela 4.4 apresenta as taxas de erro LOO-CV internas, $e(i,j,k)$, para os ternos de métodos (CB, NM, SSA) agrupadas por esquema de pesos.

		NB			sub			half		
		NN	IG-SL	IL-SL	NN	IG-SL	IL-SL	NN	IG-SL	IL-SL
InfoGain	TE	14.21	9.26	5.56	8.33	9.26	9.26	11.11	1.85	10.19
	$\#S_1$	17	62	73	34	54	71	35	60	55
	$\#S_2$	2	3	4	2	3	3	1	7	1
GiniIndex	TE	11.11	0.93	0.00	5.56	1.85	1.85	5.56	0.93	1.85
	$\#S_1$	18	117	124	36	106	129	41	106	97
	$\#S_2$	3	5	5	4	4	4	4	5	4
SVMWeighting	TE	5.56	1.85	0.00	1.85	0.93	0.93	1.85	0.00	0.933
	$\#S_1$	28	97	1014	62	88	71	67	160	153
	$\#S_2$	4	4	4	5	7	3	5	4	5
		min			edw			nexp		
		NN	IG-SL	IL-SL	NN	IG-SL	IL-SL	NN	IG-SL	IL-SL
InfoGain	TE	8.33	1.85	9.26	11.11	1.85	10.10	11.11	11.11	10.19
	$\#S_1$	34	72	71	35	59	71	22	61	63
	$\#S_2$	2	6	3	1	5	2	2	1	2
GiniIndex	TE	5.56	2.78	1.85	11.11	0.00	1.85	10.19	0.00	2.78
	$\#S_1$	37	128	129	38	101	124	26	106	112
	$\#S_2$	4	3	4	1	6	4	3	6	4
SVMWeighting	TE	1.85	2.78	0.93	0.93	0.93	1.85	4.63	0.00	0.93
	$\#S_1$	62	99	88	82	137	108	42	134	120
	$\#S_2$	5	5	5	5	3	5	4	5	6

Tabela 4.4: Taxas de erro (TE) LOO-CV (%) para o classificador MSV usando 18 combinações de métodos (CB, NM). Para cada esquema são mostrados a TE, o número de genes seleccionados após a primeira selecção através dos esquemas de pesos (S_1) e após cada um dos três procedimentos de SSA (S_2).

De modo análogo ao da Secção 4.2.2 obtiveram-se as taxa de redução (TR) devido à aplicação de uma combinação específica (CB, NM, SSA).

$$TR(i,j,k) = \frac{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA}) - e(i,j,k)}{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA})}$$

$$= \frac{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA}) - e(i,\cdot,\cdot)}{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA})} + \frac{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA}) - e(\cdot,j,\cdot)}{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA})} + \frac{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA}) - e(\cdot,\cdot,k)}{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA})}$$
$$+ \frac{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA}) - \Big(e(i,j,\cdot) - e(i,\cdot,\cdot) - e(\cdot,j,\cdot)\Big)}{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA})}$$
$$+ \frac{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA}) - \Big(e(i,\cdot,k) - e(i,\cdot,\cdot) - e(\cdot,\cdot,k)\Big)}{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA})}$$
$$+ \frac{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA}) - \Big(e(\cdot,j,k) - e(\cdot,j,\cdot) - e(\cdot,\cdot,k)\Big)}{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA})}$$
$$- \frac{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA}) - \Big(e(i,j,\cdot) + e(i,\cdot,k) + e(\cdot,j,k) - e(i,\cdot,\cdot) - e(\cdot,j,\cdot) - e(\cdot,\cdot,k) - e(i,j,k)\Big)}{e(\mathsf{NB},\mathsf{NN},\mathsf{NSSA})}$$
$$= TR(i,\cdot,\cdot) + TR(\cdot,j,\cdot) + TR(\cdot,\cdot,k) + TR(i \leftrightarrow j) + TR(j \leftrightarrow k) + TR(i \leftrightarrow k) - TR(i \leftrightarrow j \leftrightarrow k)$$

Da análise dos resultados da Tabela 4.4 e dos gráficos das Figuras 4.8 e 4.9 é possível fazer as seguintes observações:

(i) são obtidas TR superiores usando SSA em comparação com os resultados prévios (Tabela 4.2) onde não há aplicação de SSA;

(ii) relativamente à correcção de *background*, para qualquer esquema de pesos utilizado, o método de CB que origina pior desempenho preditivo é o nexp. Os métodos min e half proporcionam as TR mais elevadas por comparação com os restantes;

(iii) os métodos de CB que mostram uma TR mais reduzida sem a aplicação de SSA apresentam uma TR mais elevada quando a SSA é aplicada. Por exemplo, para os esquemas de pesos GiniIndex e SVMWeighting, o método half obtém a TR mais elevada enquanto para o InfoGain é a segunda mais elevada. Todavia, a técnica de CB half detém uma das TR mais reduzidas quando a SSA não é aplicada;

(iv) no caso da normalização não é possível tirar uma ilação muito concreta uma vez que os valores das TR são relativamente próximos para os dois métodos em estudo. No entanto, pode-se afirmar que o método IG-SL apresenta a TR mais elevada quando aplicados os esquemas de pesos InfoGain e GiniIndex, enquanto usando o es-

quema SVMWeighting é o método IL-SL.

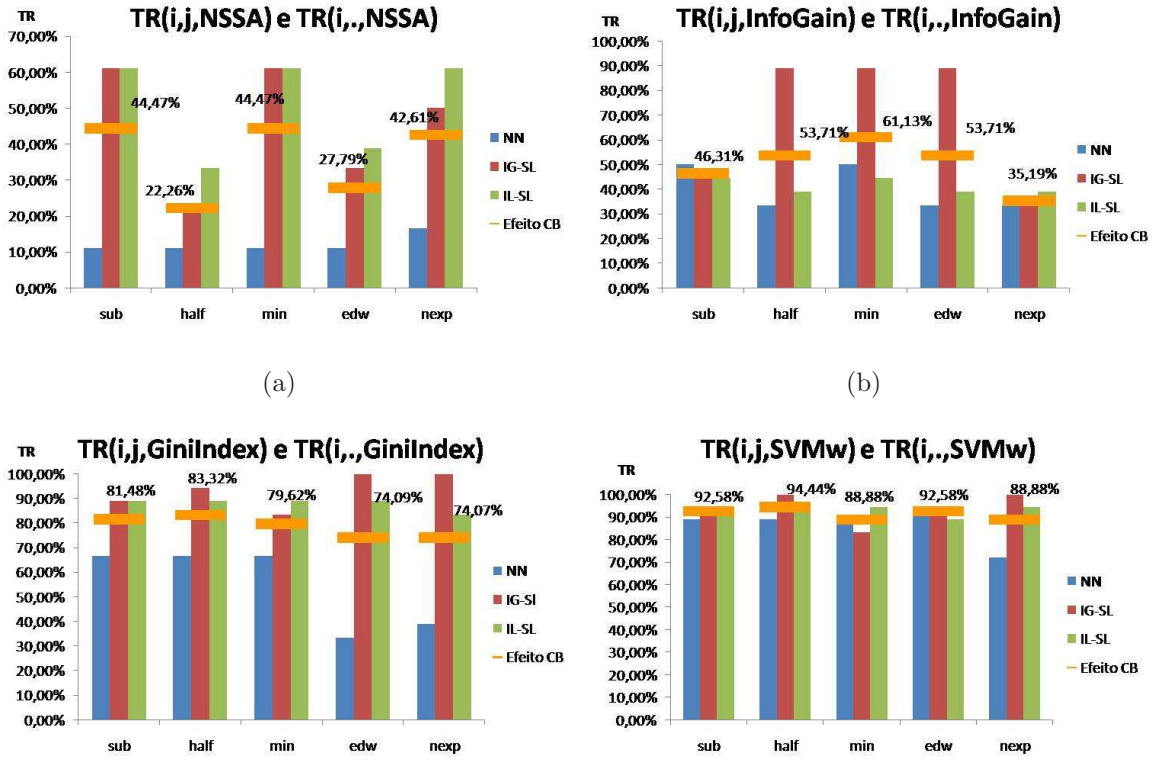


Figura 4.8: Gráficos de barras das TR por esquemas de pesos. As barras verticais representam as TR por (CB, NM, SSA) agrupadas por métodos de CB. As barras horizontais representam as TR para cada método de CB.

No sentido de extrair mais informação dos dados, após a selecção dos genes e na tentativa de averiguar informações biológicas importantes desses genes relacionadas com o cancro Linfoma foi realizado o seguinte procedimento. Após a aplicação de cada terno (CB,NM,SSA) foram guardados os genes seleccionados, na sua maioria entre 1 a 7 genes. Por cada estratégia de SSA, mais concretamente por esquema de pesos, foram registadas as ocorrências de cada gene seleccionado. Na Tabela 4.5 pode ser visualizada a lista de ordenação desses genes com base na frequência absoluta de cada um.

Depois de se ter obtido a lista dos genes com frequências absolutas mais elevadas realizou-se uma pesquisa bibliográfica constatando-se que efectivamente dois dos genes da lista, *Cyclin D2* e *Fibronectin 1*, estão de facto relacionados com o cancro Linfoma.

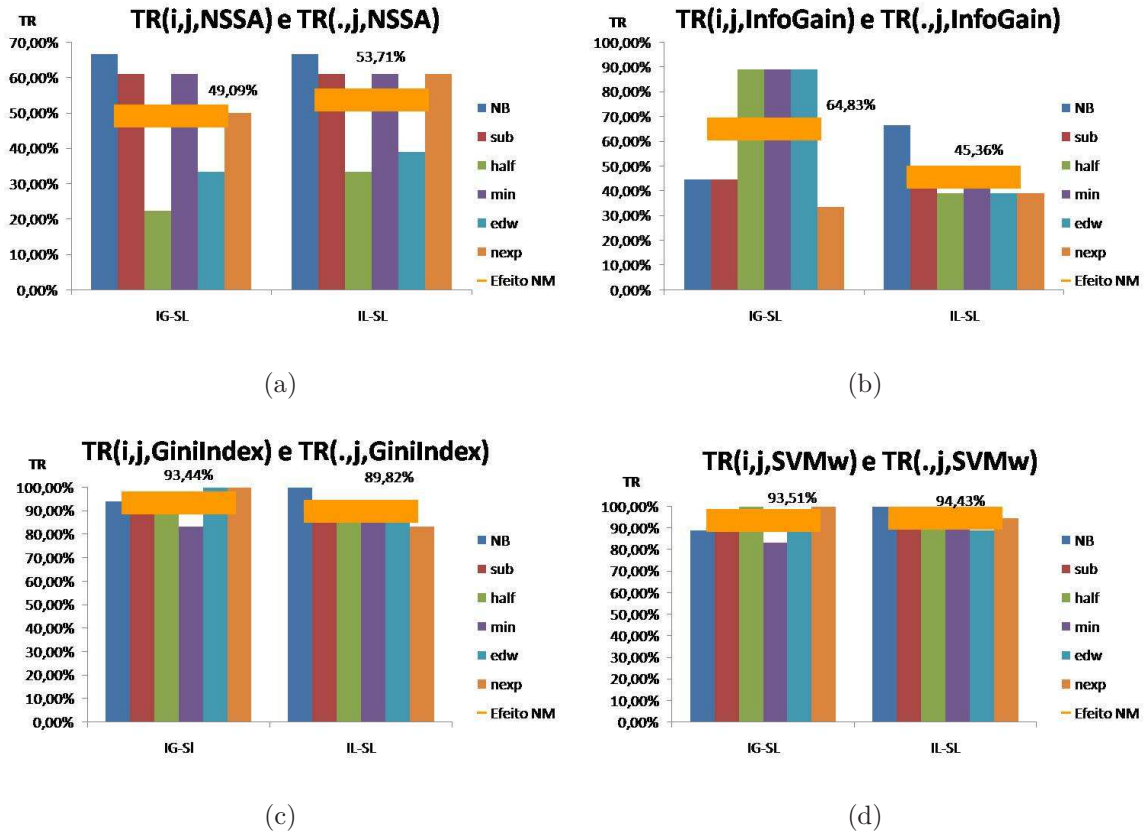


Figura 4.9: Gráficos de barras das TR por esquemas de pesos. As barras verticais representam as TR por (CB, NM, SSA) agrupadas por métodos de NM. As barras horizontais representam as TR para cada método de NM.

O gene *Cyclin D2* está envolvido na progressão das células dentro do ciclo celular enquanto que o gene *Fibronectin 1* está ligado à adesão celular, à cicatrização de ferimentos, à coagulação do sangue, às metástases, entre outras.

O estudo [29] consistiu em determinar se a identificação de subgrupos de baixo risco de desenvolvimento de *diffuse large B-cell lymphoma* (DLBCL), através de uma técnica especializada para o efeito, teria utilidade prática em relação a prognósticos e decisões terapêuticas. A identificação, na altura do diagnóstico, de pacientes com DLBCL com um prognóstico reservado pode ter um importante impacto nas decisões terapêuticas. Nesse estudo concluiu-se que as expressões de *Cyclin D2* e de *PKC-beta* estão associadas ao desenvolvimento de uma variante de DLBCL com mau prognóstico, isto é, com maior taxa de mortalidade associada. O nível de expressão do gene *Cyclin D2*, em particular,

	Ranking	Gene	# Ocurrencias	Nome do gene
InfoGain	1	63023	7	Cyclin D2
	2	140123	4	Mig=Humig=chemokine targeting T cells
	2	63146	4	Cyclin B1
	2	59870	4	AIM2
	3	15096	3	Fibronectin 1
	3	140122	3	IP-10
GiniIndex	1	62787	13	AIM2
	2	60174	7	Cyclin D2
	3	63012	5	Mitogen-activated PK 12
	4	63023	4	Protease, cysteine, 1 (legumain)
	4	60353	4	ESTs - Immunoglobulin D heavy chain constant region
	5	63146	3	Cyclin B1
	5	15096	3	Fibronectin 1
	5	62114	3	Signal transducer and activator of transcription 1, 91kD
	5	62578	3	Guanylate binding protein 1, interferon-inducible, 67kD
SVMweighting	1	62019	11	EST - Unknown UG Hs.133394
	2	62787	10	AIM2
	3	60174	7	Cyclin D2
	4	15096	3	Fibronectin 1
	4	60353	3	Immunoglobulin D heavy chain constant region
	4	64615	3	Unknown UG Hs.124890 ESTs sc_id9370
	4	67518	3	Unknown UG Hs.96731 huntingtin interacting protein-1-relat
	4	59871	3	Immunoglobulin delta 3 region

Tabela 4.5: Ordenação dos genes por frequências absolutas para cada estratégia de SSA.

está estatisticamente associado a uma redução significativa da taxa de sobrevivência.

Em [19] concluiu-se que conhecer os polimorfismos associados aos genes que codificam a *Fibronectina*, MspI e HaeIIIb, pode ser clinicamente relevante para definir o risco do desenvolvimento de DLBCL em pacientes com síndroma de crioglobulinemia. Esta é uma patologia ligada a infecção por virus da hepatite C, caracterizada por proliferação de linfócitos B e associada a um elevado risco de desenvolvimento do linfoma dito *non-Hodgkins* (NHL).

Capítulo 5

Conclusões e trabalho futuro

No presente trabalho foi realizado um estudo experimental com a finalidade de avaliar métodos combinados de correcção de *background* e de normalização com base no desempenho preditivo de dois modelos de classificação, k-vizinhos mais próximos e máquinas de suporte vectorial. Estes modelos foram induzidos de três bases de dados públicas de *microarrays* de ADN-complementar. Foi ainda executado um estudo sobre o efeito da aplicação dos métodos de CB e de NM no desempenho preditivo de classificadores de MSV quando estes são induzidos de dados constituídos apenas por subconjuntos de genes altamente discriminativos.

Os resultados apresentados neste estudo foram unicamente exploratórios recorrendo principalmente a recursos gráficos. Ao mesmo tempo foi assumido um modelo aditivo que originou o particionamento da taxa de erro LOO-CV em diferentes parcelas. Numa primeira parte do estudo experimental, esta taxa dividiu-se na contribuição do efeito de CB, na contribuição do efeito de NM e na contribuição da interacção do efeito de CB com NM. Numa segunda fase foi incluído um terceiro factor, SSA, o que levou a um modelo aditivo mais complexo mas com uma ideologia semelhante relativamente ao particionamento da taxa de erro.

Na primeira fase do estudo, ou seja, sem aplicação das técnicas de SSA, foi notório que os métodos de CB sub e nexp sobressaíram em relação aos restantes, em termos médios, ainda que através de medidas exploratórias. O resultado do método nexp vem ao encontro do obtido em [53]. Todos os recursos envolvidos *(i)* taxas de redução do erro LOO-

CV; *(ii)* diferenças positivas entre os métodos base e cada combinação de métodos, e a *(iii)* medição da contribuição do efeito de cada método de pré-processamento tendo em conta apenas a taxa de erro LOO-CV, corroboraram as conclusões de cada um dos recursos usados para comparar os métodos de pré-processamento estudados na presente dissertação. Em relação aos métodos de NM, destacou-se o método de 2-passos IL-SL que confirma os resultados obtidos em [76].

Os resultados não se verificaram os mesmos aquando da segunda fase deste estudo experimental, isto é, quando houve a aplicação de técnicas de SSA. Os métodos de CB que se destacaram no estudo prévio não se salientaram neste segundo, aliás, houve uma inversão de comportamento, ou seja, os métodos que estavam associados a uma taxa de erro mais reduzida no primeiro estudo tornaram-se métodos associados a uma taxa de erro mais elevada no segundo. Ao nível da normalização, os resultados apontam o método IG-SL como sendo aquele com uma taxa de redução mais elevada.

O segundo estudo revelou ser bastante elucidativo na selecção dos genes que estavam mais relacionados com o cancro Linfoma. Através de uma pesquisa bibliográfica foi possível obter evidências que dois dos genes com maior frequência absoluta, em termos de serem seleccionados pelas técnicas de SSA, estão de facto implicados no aparecimento desse tipo de cancro.

Em termos concretos, para a área da Bioinformática, os resultados exploratórios obtidos para os métodos de CB sub e nexp, avaliados com base na capacidade preditiva dos classificadores induzidos neste estudo, permitem dizer que são esses os que apresentam taxas de redução mais elevadas. Portanto, e apenas com base nesta análise, é possível indicá-los como método de CB a usar em detrimento dos restantes. No caso dos métodos de NM, é possível afirmar, com base nas medidas exploratórias, que os métodos mais indicados a aplicar aos dados são os de 2-passos IG-SL e IL-SL.

As publicações feitas no âmbito desta dissertação foram duas: [23] e [55]. As principais contribuições desta dissertação foram apresentadas em duas conferências donde resultaram essas duas publicações. A primeira foi apresentada no XVI Congresso da SPE - UTAD realizado em Vila real em 2008 e a segunda na conferência de Artificial Intelligence in Medicine realizada em Verona em 2009.

Apresenta-se como linha de trabalho futuro a investigação de métodos combinados

de correcção de *background* e de normalização com base no desempenho preditivo de modelos de classificação onde se tenha em conta a variância da taxa de erro LOO-CV. Seria também útil fazer testes estatísticos que permitissem avaliar a significância dos efeitos dos métodos de pré-processamento detectados na análise exploratória de dados aqui efectuada. Outro enriquecimento científico que podia ser dado a este trabalho passa, por um lado, pela utilização de mais bases de dados e, por outro lado, pelo uso de bases de dados com um número m de tecidos mais elevado das que aqui se utilizaram. Os resultados obtidos e as questões deixadas em aberto podem constituir uma boa base de trabalho futuro.

Bibliografia

[1] Amaratunga D. and Cabrera J., *Exploration and analysis of DNA microarray and protein array data*, John-Wiley & Sons, Inc, USA, 2004.

[2] Ambroise C. and McLachlan G.J., *Selection Bias in Gene Extraction on the Basis of Microarray Gene-Expression Data*, Proc. Natl Academy of Sciences USA, Vol. 99, No. 10, pp. 6562-6566, 2002.

[3] Bolstad B.M., Irizarry R.A., Astrand M. and Speed T.P., *A comparison of normalization methods for high density oligonucleotide array data based on variance and bias*, Vol. 19, No. 2, pp. 185-103, 2003.

[4] Bolstad B.M., *Low Level Analysis of High-density Oligonucleotide Array Data: Background, Normalization and Summarization*, PhD Dissertation, Department of Biostatistics, University of California, Berkeley, 2004.

[5] Brazma A., Vilo J. and Cesareni E.G., *Gene Expression Data Analysis*, FEBS Lett, Vol. 480, pp. 17-25, 2000.

[6] Brown M.P.S., Grundy W.N., Lin D., Cristianini N., Sugnet C., Ares M. and Haussler D., *Support Vector Machine Classification of Microarray Gene Expression Data*, Technical Report UCSC-CRL-99-09, University of California, Santa Cruz, CA, 1999.

[7] Buckley M.J., *The Spot user's guide, CSIRO Mathematical and Information Sciences*, 2000. Available online at http://www.cmis.csiro.au/IAP/Spot/spotmanual.htm .

[8] Castillo G., *Adaptive Learning Algorithms for Bayesian Network Classifier*, PhD thesis, Department of Mathematics, University of Aveiro, 2006.

[9] Causton H.C., Quackenbush J., and Brazma A., *Microarray Gene Expression Data Analysis: A Beginner's Guide*, Blackwell Publishing, 2003.

[10] Chen Y., Dougherty E.R. and Bittner M.L., *Ratio-based decisions and the quantitative analysis os cDNA microarray images*, Journal of Biomedical optics, Vol. 2, pp. 364-374, 2000.

[11] Chen Z., McGee, M., Qingzhong L., Kong M., Youping D. and Scheuermann R.H., *A distribution-free convolution model for background correction of oligonucleotide microarray data*, BMC Genomics, 10(Suppl 1):S19, 2009.

[12] Chih-Chung C. and Chih-Jen L., *LIBSVM: a library for support vector machines*, 2001. Acessível on-line em http://www.csie.ntu.edu.tw/ cjlin/libsvm .

[13] Cleveland W.S., *Robust Locally Weighted Regression and Smoothing Scatterplots*, Journal of the American Statistical Association, Vol. 74, No. 368, pp. 829-836, 1979.

[14] Cleveland W.S. and Devlin S.T., *Locally Weighted Regression: An Approach to Regression Analysis by Local Fitting*, Journal of the American Statistical Association, Vol. 83, No. 403, pp. 596-610, 1988.

[15] Cleveland W.S., Grosse E. and Shyu M.J., *A package of C and Fortran routines for fitting local regression models*, August 1992.

[16] Dietterich T.G. and Bakiri G., *Error-correcting output codes: A general method for improving multiclass inductive learning programs*. In Proceedings of AAAI-91, AAAI Press, pp. 572-577, 1991.

[17] Dudoit S., Yang Y.H., Callow M.J. and Speed T.P., *Statistical methods for identifying differentially expressed genes in replicated cDNA microarrays experiments*, Statistica Sinica, Vol. 12, No. 1., pp. 111-139, 2002.

[18] Edwards D., *Non-linear normalization and background correction in one-channel cDNA microarray studies*, Bioinformatics, Vol. 19, No. 7, pp. 825-833, 2003.

[19] Fabris M., Quartuccio L., Salvin S., Pozzato G., De Re V., Mazzaro C., Ferri C., Baldini C. and De Vita S., *Fibronectin gene polymorphisms are associated with the development of B-cell lymphoma in type II mixed cryoglobulinemia*, Annals of the Rheumatic Diseases, 67:80-83, 2008.

[20] Fang H., Fan X., Guo L., Shi L., Perkins R., Ge W., Dragan Y.P. and Tong W., *Self-self Hybridization As An Alternative Experiment Design to Dye Swap for Two-color Microarrays*, OMICS: A Journal of Integrative Biology, Vol. 11, No. 1, pp. 14-24, 2007.

[21] Fayyad U.M., Piatetsky-Shapiro G., Smyth P. and Uthurusamy R., *Advances in knowledge discory and data mining*, American Association for Artificial Inteligence (AAAI), USA, 1996.

[22] Finkelstein D.B., Gollub J., Ewing R., Sterky F., Somerville S. and Cherry J.M., *Iterative linear regression by sector*, In: Methods of Microarray Data Analysis, Papers from CAMDA 2000. Edited by: Lin S.M. and Johnson K.F., Kluwer Academic, pp. 57-68, 2001.

[23] Freitas A., Castillo G., São Marcos A.L., *Effect of background correction on cancer classification with gene expression data*, Proceedings of the AIME'09, Artificial Intelligence in Medicine, Lecture Notes in Artificial Intelligence, Springer Verlag, pp. 416-420, 2009.

[24] Freudenberg J.M., *Comparison of background correction and normalization procedures for high-density oligonucleotide microarrays*, Leipzig Bioinformatics Working Paper No. 3, Universität Leipzig, January 2005.

[25] *GenePix 4000B Users Guide*, Axon Instruments, Inc., 2500-136 Rev E, 2001. Acessível online em http://www.moleculardevices.com/home.html .

[26] Golub T.R., Slonim D.K., Tamayo P., Huard C., Gaasenbeek M., Mesirov J.P., Coller H., Loh M.L., Downing J.R., Caligiuri M.A. and Bloomfield C.D., *Molecular classification of cancer: class discovery and class prediction by gene expression monitoring*, Science, Vol. 286, pp. 531537, 1999.

[27] Grosso A.F., *Statistical Methodologies for the Analysis of DNA Microarray Data*, Mestrado em Bioinformática, Faculdade de Ciências da Universidade de Lisboa, 2006.

[28] Guyon I., Weston J., Barnhill S. and Vapnik V., *Gene Selection for Cancer Classification using Support Vector Machines*, Machine Learning, Vol. 46, No. 1, pp. 389-422, 2002.

[29] Hans C.P., Weisenburger D.D., Greiner T.C., Chan W.C., Aoun P., Cochran G.T., Zenggang P., Smith L.M., Lynch J.C., Bociek R.G., Bierman P.J., Vose J.M. and Armitage J.O., *Expression of PKC-beta or cyclin D2 predicts for inferior survival in diffuse large B-cell lymphoma*, Modern Pathology, Vol. 18, pp. 13771384, 2005.

[30] Hartemink A.J., Gifford D.K., Jaakkola T.S. and Young R.A., *Maximum likelihood estimation of optimal scaling factors for expression array normalization*, Proc. Int'l Symp. Biomedical Optics, 2001.

[31] Hastie T.J. and Tibshirani R.J., *Classification by pairwise coupling*, Advances in Neural Information Processing Systems, Editied by: Jordan M.I., Kearnsa M.J., Solla S.A., MIT Press, Vol. 10, pp. 507-513, 1998.

[32] Ihaka R. and Gentleman R., *R: a language for data analysis and graphics*, Journal of Computational and Graphical Statistics, Vol. 5, pp. 299-314, 1996.

[33] Irizarry R.A., Hobbs B., Collin F., Beazer-barclay Y.D., Antonellis K.J., Scherf U. and Speed T.P., *Exploration, Normalization, and Summaries of High Density*, Bioestatistics, Vol. 4, No. 2, pp. 249-264, 2003.

[34] Jiawei H. and Kamber M., *Data mining - Concepts and Techniques*, 2nd Edition, Morgan Kaufmann Publishers, 2006.

[35] Karush W., *Minima of Functions of Several Variables with Inequalities as Side Constraints*, M.Sc. Dissertation, Department of Mathematics, University of Chicago, Illinois, 1939.

[36] Kepler T.B., Crosby L. and Morgan K.T., *Normalization and analysis of DNA microarray data by self-consistency and local regression*, Santa Fe Institute Working Paper, Santa Fe, New Mexico, 2000.

[37] Kuhn H.W. and Tucker A.W., *Nonlinear programming*, Proceedings of 2nd Berkeley Symposium, Berkeley: University of California Press, pp. 481-492, 1951.

[38] Kooperberg C., Fazzio T.G., Delrow J.J. and Tsukiyama T., *Improved background correction for spotted DNA microarrays*, Journal of Computational Biology, Vol. 9, No. 1, pp. 5566, 2002.

[39] Kroll T.C. and Wölfl S., *Ranking: a closer look on globalisation methods for the normalization of gene expression arrays*, Nucleic Acids Res, 30, e50, 2002.

[40] Lipo W., Feng C., and Wei X., *Accurate Cancer Classification Using Expressions of Very Few Genes*, IEEE/ACM Trans. Comput. Biology Bioinform, Vol. 4, No. 1, pp. 40-53, 2007.

[41] Lipshutz R.J., Fodor S.P., Gingeras T.R. and Lockhart D.J., *High density synthetic oligonucleotide arrays*, Nature Genetics, Vol. 21, No.1, pp. 20-24, 1999.

[42] Lockhart D.J., Dong H., Byrne M.C., Follettie M.T., Gallo M.V., Chee M.S., Mittmann M., Wang C., Kobayashi M., Horton H. and Brown E.L., *Expression monitoring by hybridization to high-density oligonucleotide arrays*, Nature Biotechnology, Vol. 14, No. 13, pp. 16751680, 1996.

[43] Lopes S., *Técnicas Geométricas de Condensação para o Classificador k-NN*, Tese de Mestrado, Departamento de Matemática, Universidade de Aveiro, 2008.

[44] Mierswa I., Wurst M., Klinkenberg R., Scholz M. and Euler T., *YALE: Rapid Prototyping for Complex Data Mining Tasks*, in Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-06), pp.935-940, 2006.

[45] Mitchell T., *Machine Learning*, McGraw-Hill, 1997.

[46] Mukherjee S., *Classifying Microarray Data Using Support Vector Machines* - Chapter 9, Edited by: Berrar D.P., Dubitzky W., Granzow M., A practical approach to microarray data analysis, Kluwer Academic Publishers, 2003.

[47] Pang-Ning T., Steinbach M. and Kumar V., *Introduction to data mining*, Adison-Wesley, 2006.

[48] Park T., Yi SG., Kang SH., Lee S., Lee YS. and Simon R., *Evaluation of normalization methods for microarray data*, BMC Bioinformatics, 4:33, 2003.

[49] *QuantArray Microarray Analysis Software Manual*, Packard BioScience, USA, 2001. Available online at http://las.perkinelmer.com/content/manuals/man_quantarraysoftware.pdf .

[50] Quin L. and Kerr F., *Empirical evaluation of data transformations and ranking statistics for microarray analysis*, Nucleic Acids Research, Vol. 32, No. 18, pp. 5471-5479, 2004.

[51] Quinlan J.R., *C4.5: Programs for Machine Learning*, Morgan Kaufmann Publishers, Inc., San Mateo, California, 1993

[52] Ritchie M.E., *Quantitative quality control and background correction for two-colour microarray data*, Ph.D. thesis, Department of Medical Biology, University of Melbourne, 2004.

[53] Ritchie M.E., Silver J., Oshlack A., Holmes M., Diyagama D., Holloway A., and Smyth G.K., *A comparison of background correction methods for two colour microarrays*, Bioinformatics, Vol. 23, No. 20, pp. 2700-2707, 2007.

[54] Rocha M., Cortez P. and Neves J.M, *Análise Inteligente de Dados - Algoritmos e Implementação em JAVA*, FAC - Editora de Informática, 2008.

[55] São Marcos A., Freitas A., Castillo G., *Avaliação de métodos de correcção de background e normalização em dados de microarrays de ADN-complementar*, Livro de actas do XVI Congresso da SPE, UTAD - Vila REAL, 2008.

[56] Scharpf R.B., Iacobuzio-Donahue C., Sneddon, J.B. and Parmigiani G., *When should one subtract background fluorescence in 2-color microarrays?*, Biostatistics, Vol. 8, No. 4, pp. 695-707, 2007.

[57] Schena M., Shaon D., Heller R., Chai A., Brown P. and Davis R, *Parallel human genome analysis: microarray-based expression monitoring of 1000 genes*, Proc. Natl Academy of Sciences USA, Vol. 93, No. 20, pp. 1061410619, 1996.

[58] Shi L., Tong W., Su Z., Han T., Han J., Puri R.K., Fang H., Frueh, F.W., Goodsaid F.M., Guo L., Branham W.L., Chen J.J., Xu Z.A., Harris S.C., Huixiao H., Qian X., Perkins R.G. and Fuscoe J.C, *Microarray scanner calibration curves: characteristics and implications*, Bioinformstics, 6(Suppl 2): S11, 2005.

[59] Sierra B., *Aportaciones metodológicas a la Clasificación Supervisada*, Tesis Doctoral, Departamiento Ciencias de la Computación e Inteligencia Artificial, Universidad del País Vasco, 2000.

[60] Smyth G.K., Yang Y.H. and Speed T.P., *Statistical issues in cDNA microarray data analysis*, In Functional Genomics: Methods and Protocols Vol. 224. Edited by: Brownstein M.J., Khodursky A.B., Totowa N.J., Humana Press Vol. 224, pp. 111-136, 2003.

[61] Smyth G.K. and Speed T.P., *Normalization of cDNA microarray data*, In: Methods - Selecting Candidate Genes from DNA Array Screens: Application to Neuroscience, Edited by: Carter D., pp. 265-273, 2003.

[62] Smyth G.K., *Linear models and empirical Bayes methods for assessing diFFerential expression in microarray experiments*, Statistical Applications in Genetics and Molecular Biology, Vol. 1, No. 3, 2004.

[63] Stanford Microarray Database. Acessível online em http://genome-www5.stanford.edu/ .

[64] The R/Bioconductor package. Acessível online em http://www.bioconductor.org/ .

[65] Tusher V.G., Tibshirani R. and Chu G., *Significance analysis of microarrays applied to the ionizing radiation response*, Proc. Natl Academy of Sciences USA, Vol. 98, pp. 51165121, 2001.

[66] Vapnik V.N., *Statistical Learning Theory*, Wiley Interscience, New York, 1998.

[67] Yang Y.H., Buckley M.J. and Speed T.P, *Analysis of cDNA microarray images*, Briefings in Bioinformatics, Vol. 2, No. 4, pp. 341-349, 2001.

[68] Yang Y.H., Dudoit S., Luu P. and Speed T.P., *Normalization for cDNA microarray data*, San Jose, California, Vol. 4266. Edited by: Bittner M.L., Chen Y., Dorsel A.N. and Dougherty E.R., SPIE-International Society for Optical Engineering, pp. 141-152, 2001.

[69] Yang Y.H., Dudoit S., Luu P., Lin D.M., Peng V., Ngai J. and Speed T.P, *Normalization for cDNA microarray data: a robust composite method addressing single and multiple slide systematic variation*, Nucleic Acids Research, Vol. 30, No. 4, 2002.

[70] Yang Y.H., Buckley M.J., Dudoit S., and Speed T.P., *Comparison of methods for image analysis on cDNA microarray data*. Journal of Computational and Graphical Statistics, Vol. 11, No. 1, pp. 108-136, 2002.

[71] Yang Y.H., Xiao Y. and Segal M.R., *Identifying differentially expressed genes from microarray experiments via statistic synthesis*, Bioinformatics, Vol. 21, No. 7, pp. 1084-1093, 2004.

[72] Ying L. and Jiawei H., *Cancer classification using gene expression data*, Information Systems, Vol.28, pp. 243-268, 2003.

[73] Wilson D.L., Buckley M.J., Helliwell C.A. and Wilson I.W., *New normalization methods for cDNA microarray data*. Bioinformatics, Vol. 19, No. 11, pp. 1325-1332, 2003.

[74] Witten I.H. and Frank E., *Data Mining: Practical Machine Learning Tools and Tech- niques with Java Implementations*, Morgan Kaufmann Publishers, Inc, 1999.

[75] Workman C., Jensen L.J., Jarmer H., Berka R., Gautier L., Nielser H.B., Saxild H.H., Nielsen C., Brunak S. and Knudsen S., *A new non-linear normalization method for reducing variability in DNA microarray experiments*, Genome Biol, Vol. 3, No. 9., 2002.

[76] Wu W., Xing E., Myers C., Mian I.S. and Bissel M.J., *Evaluation of normalization methods for cDNA microarray data by k-NN classification*, BMC Bioinformatics, 6:191, 2005.

[77] http://www.itl.nist.gov/div898/handbook/pmd/section1/pmd144.htm. Sítio visitado em 07/10/09.

[78] http://array.bioengr.uic.edu/∼yangdai/teach/bioe594-spring03/lowess.pdf . Sítio visitado em 07/10/09.

[79] http://www.ncbi.nlm.nih.gov/Class/NAWBIS/Modules/Expression/exp1.html . Sítio visitado em 28/09/09.